SOFT COMPUTING BASED SPATIAL ANALYSIS OF EARTHQUAKE TRIGGERED COHERENT LANDSLIDES

A Dissertation
Presented to
The Academic Faculty

by

Mesut Turel

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Civil and Environmental Engineering

Georgia Institute of Technology
December 2011

SOFT COMPUTING BASED SPATIAL ANALYSIS OF EARTHQUAKE TRIGGERED COHERENT LANDSLIDES

Approved by:

Dr. J. David Frost, Advisor
School of Civil and Environmental Engineering
Georgia Institute of Technology

Dr. Hermann Fritz
School of Civil and Environmental Engineering
Georgia Institute of Technology

Dr. Glenn J. Rix
School of Civil and Environmental Engineering
Georgia Institute of Technology

Dr. Steven P. French
College of Architecture
Georgia Institute of Technology

Dr. Yi-Chang James Tsai
School of Civil and Environmental Engineering
Georgia Institute of Technology

Date Approved: November 7, 2011

To My Family and My Wife

ACKNOWLEDGEMENTS

My sincerest gratitude goes to my advisor Dr. David Frost. His support, guidance, friendship and generosity made this journey much more enjoyable. I feel very lucky to have had the opportunity to work with him. I couldn't have asked for a better mentor.

I would like to show appreciation to Dr. Glenn J. Rix, Dr. Yi-Chang James Tsai, Dr. Hermann Fritz, and Dr. Steven P. French for serving as my committee members. Their time and comments are greatly appreciated.

I would like to express thanks to all my good friends in Savannah, Zafer Defne, Fahad Mohammed, Elza Bystrom, Ye Lu, Kemal Cambazoglu, Huseyin Demir, Chad Spurlock, Chatchawin Srisuwan, and Xiufeng Yang.

I would like to thank to my parents Fikriye and Celal Turel, my sister Zeynep and my brother Harun for their love and moral support.

Last but not least many thanks to my lovely wife Luz Karina Turel for her love and support.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

Earthquake triggered landslides cause loss of life, destroy structures, roads, powerlines, and pipelines and therefore they have a direct impact on the social and economic life of the hazard region. The damage and fatalities directly related to strong ground shaking and fault rupture are sometimes exceeded by the damage and fatalities caused by earthquake triggered landslides. Even though future earthquakes can hardly be predicted, the identification of areas that are highly susceptible to landslide hazards is possible. For geographical information systems (GIS) based deterministic slope stability and earthquake-induced landslide analysis, the grid-cell approach has been commonly used in conjunction with the relatively simple infinite slope model. The infinite slope model together with Newmark's displacement analysis has been widely used to create seismic landslide susceptibility maps. The infinite slope model gives reliable results in the case of surficial landslides with depth-length ratios smaller than 0.1. On the other hand, the infinite slope model cannot satisfactorily analyze deep-seated coherent landslides. In reality, coherent landslides are common and these types of landslides are a major cause of property damage and fatalities. In the case of coherent landslides, two- or three-dimensional models are required to accurately analyze both static and dynamic performance of slopes. These models are rarely used in GIS-based landslide hazard zonation because they are numerically expensive compared to one dimensional infinite slope models. Building metamodels based on data obtained from computer experiments and using computationally inexpensive predictions based on these metamodels has been widely used in several engineering applications. With these soft computing methods,

design variables are carefully chosen using a design of experiments (DOE) methodology to cover a predetermined range of values and computer experiments are performed at these chosen points. The design variables and the responses from the computer simulations are then combined to construct functional relationships (metamodels) between the inputs and the outputs. In this study, Support Vector Machines (SVM) and Artificial Neural Networks (ANN) are used to predict the static and seismic responses of slopes. In order to integrate the soft computing methods with GIS for coherent landslide hazard analysis, an automatic slope profile delineation method from Digital Elevation Models is developed. The integrated framework is evaluated using a case study of the 1989 Loma Prieta, CA earthquake (Mw = 6.9). A seismic landslide hazard analysis is also performed for the same region for a future scenario earthquake (Mw = 7.03) on the San Andreas Fault.

CHAPTER 1

INTRODUCTION

1.1. Motivation for Research

Earthquake-triggered landslides and slope failures represent major seismic hazards and pose a significant threat to both human life and property. It is well documented that earthquake-triggered landslides have caused hundreds of thousands of deaths and substantial economical losses. The main objective of regional earthquake-triggered landslide hazard analysis is to evaluate the location of the areas where landslides can be triggered by future earthquakes. The susceptibility of an area to earthquake-triggered landslides can be assessed based on the potential ground motion, and geological and geomorphological conditions.

The Geographic Information System (GIS) provides a powerful set of tools for collecting, storing and analyzing spatial data for regional earthquake-triggered landslide hazard analysis. In the last two decades, GIS technology has been extensively used in developing process-driven models of seismic slope deformation (Miles and Ho, 1999; Jibson et al., 2000; Khazai, 2004). Almost all of these methods use the infinite slope model which consists of a simple equation; therefore these methods can be easily implemented in GIS. Such methods are particularly relevant when assessing earthquake triggered shallow landslides. However, infinite slope models cannot analyze deep-seated coherent landslides (Mankelow, 1997; Xie et al., 2003; Luo, 2006; Baum et al., 2008). There is a research need for integrating seismic slope stability analysis for deep-seated coherent landslides with GIS for a regional landslide hazard analysis. This research need is addressed in this study through the use of soft computing methods for seismic stability

of slopes and integrating the soft computing methods with automated slope profile delineation methods.

The overall objective of this research is the development of a framework for the evaluation of earthquake-triggered coherent landslide hazards. This framework will consist of reliable metamodels based on the data obtained from computationally expensive static and dynamic slope stability analyses and using computationally inexpensive predictions of slope performances during future events based on these metamodels. Additionally, in order to use these metamodels for a regional landslide hazard analysis there is a need for an automated slope profile delineation method so that the slope geometry can be extracted and entered into the model as an input.

1.2. Organization of the Thesis

The thesis is composed of 7 chapters. The chapters are organized in the following manner:

- Chapter 2 gives a brief overview of earthquake-induced landslide characteristics and analysis methods. It describes different classes of landslides and their properties. It presents different analytical methods for analyzing the seismic stability of slopes. It also provides general information on application of GIS to regional landslide hazard zonation.
- Chapter 3 provides detailed information and relevant background on soft computing methods. It describes different metamodeling techniques such as Response Surface Methodology (RSM), Artificial Neural Networks (ANN), Support Vector Machines (SVM) and Naïve Bayes Classifiers. It presents Design of Experiments (DOE) methodologies including Latin Hypercube Designs (LHS). This chapter also provides a feasibility study to explore the effectiveness of metamodeling process for approximating deterministic computer experiments. The static factor of safety calculation of a homogenous

dry slope is used as a relatively simple, yet realistic example to demonstrate the steps involved in metamodeling and familiarize the reader with the process of creating metamodels. The performances of different metamodeling techniques are compared.

- Chapter 4 presents the use of soft computing methods for predicting seismic responses of slopes. In the first part, it starts with application of ANN regression metamodels for the case of a single slope under different earthquake loads. At first, it uses a sequential Latin Hypercube sampling for the design of experiments using moment magnitude of the earthquake and distance from the epicenter as design variables and generating artificial earthquake records using a stochastic point-source model. After that, it investigates the performance of the trained metamodel on experiments with real earthquake records. Next, the problem is modeled as a classification problem and SVM classifiers are used as metamodels. In the second part, SVM classifiers are used to predict seismic performances of different slopes under different earthquake loads. Addition to moment magnitude and distance from epicenter, six parameters (design variables) are chosen for the geometric and material properties of the slopes. Both real and artificial earthquake records are used and the performances of the metamodels are investigated.
- Chapter 5 introduces alternative methodologies to delineate two dimensional slope profiles from a given Digital Elevation Model (DEM). An extension to these methodologies is also presented. This extension can be used specifically to delineate slope profiles along highway sections.
- Chapter 6 presents an integrated framework that is developed for combining the metamodels described in Chapter 3 for predicting the static factor of safety and in Chapter 4 for predicting the performance of slopes under dynamic loading with the automated slope profile delineation method introduced in

Chapter 5. The integrated framework is evaluated using a case study of the 1989 Loma Prieta earthquake (Mw = 6.9) within the 7.5 min. Laurel, CA quadrangle. A seismic landslide hazard analysis is also performed for the same region for a future scenario earthquake (Mw = 7.03) on the San Andreas Fault.

- Chapter 7 summarizes the findings of the present study and provides some recommendations for future work.

CHAPTER 2

EARTHQUAKE-TRIGGERED LANDSLIDE CHARACTERISTICS AND ANALYSIS METHODS

2.1. Introduction

12 May 2008 started as an ordinary school day for the 500 students and teachers in Xinbei Middle School in Beichuan County, China. At 14:28 local time, a M_w 7.9 Wenchuan earthquake hit Sichuan Province. One of the 15,000 landslides triggered by the earthquake with a volume of 2.4 million m^3 buried Xinbei Middle School with its students and teachers under landslide debris. Concurrently, on the west side of the town, Chengxi landslide caused 1,600 deaths and destroyed half of the old area of the Beichuan County Town. Landslides and rockfalls killed 3,807 people in Beichuan County (see Figure 2.1). In the Wenchuan earthquake, 20,000 out of 68,636 total confirmed deaths were caused by landslides, rockfalls and debris flows (Wang et al., 2009; Wang et al., 2009).

Earthquake-induced landslides have been documented from as early as 18th century BCE in China and 3rd century BCE in Greece (Keefer, 2002). There are numerous examples of landslides in the recent history which buried whole cities and killed thousands of people in the matter of seconds. Early in the 20th century, in 1920, Haiyuan earthquake in northwest China with moment magnitude M_w=7.8 triggered thousands of landslides that killed over 100,000 people (Keefer, 1984; Luo, 2006). The Peruvian earthquake in 1970 caused numerous large scale landslides and one of which was the most destructive single earthquake-induced landslide in historic time that killed almost 25,000 people and buried the villages of Yungay and Ranrahirca in Santa Valley (Bommer et al., 1999; Keefer, 2002; Keefer and Larsen, 2007). The fatalities caused by slope hazards were more than 50% of the total deaths in several earthquakes that occurred between 1964 and 1978 in Japan (Kobayashi, 1981). The first attempt to map all of the

landslides produced by a large earthquake was the landslide inventory prepared by Harp et al. (1981) after the 1976, Guatemala earthquake (M=7.5). Aerial photographs and fields studies revealed more than 10,000 landslides (Harp et al., 1981). One large landslide triggered by the 2001 El Salvador earthquake (M=7.6), the Las Colinas landslide, caused 585 fatalities (Jibson et al., 2004). The 1999 Chi-Chi earthquake triggered more than 10,000 landslides throughout an area about 11,000 km^2 (Khazai and Sitar, 2004).

North America, specifically California, has a history of fatal and destructive landslides triggered by earthquakes. The great San Francisco earthquake in 1906, with a moment magnitude M_w=7.9, triggered more than 10,000 landslides and these landslides killed at least 11 people and caused extensive damage to structures, roads and railroads (Keefer, 1998). The 1989 Loma Prieta earthquake in central California created thousands of landslides throughout the region. They damaged at least 200 residences, numerous roads and other structures and caused one fatality. The direct damage from the landslides exceeded $30 million (Keefer, 1998). The Loma Prieta earthquake provided scientists with the first relatively complete data set on landslides generated by an earthquake of this size in central California. The 17 January, 1994 Northridge, California earthquake (M_w=6.7) triggered more than 11,000 landslides over an area of about 10,000 km^2 (Harp and Jibson, 1996). The Northridge earthquake produced one of the most comprehensive data sets of strong ground motion, geologic effects and landslide inventories (Parise and Jibson, 2000). California is not the only region in North America that is prone to earthquake triggered landslide hazards. The 1964 Alaskan earthquake with M_w=9.2 provided much data on landslides even though the affected area was very large and complete landslide inventory was not attempted (Keefer, 2002). The majority of the $0.67 billion damage was caused by landslides (Mankelow, 1997). Numerous landslides were triggered along the bluffs bordering the Mississippi alluvial plain in western

Tennessee and Kentucky during the New Madrid earthquakes of 1811-1812 (Jibson and Keefer, 1993).

Landslides cause loss of life, destroy structures, roads, powerlines, lifelines therefore they have a direct impact on the social and economic life of the hazard region. The damage and fatalities directly related to strong ground shaking and fault rupture are sometimes exceeded by the damage and fatalities caused by earthquake triggered landslides (Keefer, 1984). In recent decades, population growth and expansion of cities and life-lines over areas prone to natural hazards increased the impact of natural disasters such as landslides both in developed and undeveloped countries (Guzzetti et al., 1999). Even though future earthquakes can hardly be predicted, the identification of areas that are highly susceptible to landslide hazards is possible. Any comprehensive seismic hazard assessment must include hazards associated with earthquake triggered landslides (Bommer et al., 1999). At present, landslide research efforts around the globe are generally small compared to the costs of landslide damage. A recent report by the U.S. National Research Council recommended a 15-fold increase in funding for landslide research and development in the United States (Keefer and Larsen, 2007).

Figure 2.1 Town of Beichuan (photo taken two years after the earthquake on May 2010)

2.2. Classification of Earthquake Triggered Landslides

The term *landslide* refers to "the movement of a mass of rock, debris or earth down a slope" (Turner and Schuster, 1996). Keefer (1984) extended this definition to include lateral movement of earth materials as well as downslope movements. Different landslide classifications are proposed based on material, mechanism of initiation, character of movement and morphology (Varnes, 1978). Keefer (1984) studied 40 historical earthquakes worldwide and identified fourteen different types of earthquake-triggered landslides based on the principles and terminology of Varnes (1978). Table 2.1 shows fourteen types of landslides identified by Keefer (1984), with their corresponding type of movement, minimum local earthquake magnitude (M_L) required for generating the landslide, relative abundance of the landslide type, minimum slope angle required for landslide initiation and the depth of the landslide. The fourteen types of landslides are also categorized into three main categories as, disrupted landslides, coherent landslides, and lateral spreads and flows. The number of documented landslides produced by earthquakes with respect to these three main categories can be seen in Table 2.2. Keefer (1984) classifies the material "soil" or "rock" on the basis of its condition prior to the landslide initiation. "Soil" defines a loose, unconsolidated, or poorly cemented aggregate material. The term includes the entire regolith and man-made fills. The term "rock" defines firm, intact bedrock.

Three detailed landslide inventories are available in the literature for the 1999 Chi-Chi, 1994 Northridge and the 1989 Loma Prieta earthquakes (Harp and Jibson, 1996; Keefer, 1998; Keefer, 2000; Parise and Jibson, 2000; Khazai and Sitar, 2004). These detailed inventories provide the opportunity to evaluate the characteristics of the different types of landslides under different earthquake and geological conditions. Next, the different types of landslides will be investigated in more detail and investigation will be supplemented by examples from these three earthquakes.

Table 2.1 Characteristics of earthquake-triggered Landslides (Modified from Keefer (1984))

	Name	Type of Movement	Magnitude	Abundance	Slope	Depth
Disrupted Landslides	Rock Falls	Bouncing, rolling, free fall	$M_L \geq 4.0$	Very Abundant	$> 40^o$	Shallow
	Disrupted Rock Slides	Translational sliding	$M_L \geq 4.0$	Very Abundant	$> 35^o$	Shallow
	Rock Avalanches	Complex, involving sliding, flow, and occasionally free fall	$M_L \geq 6.0$	Uncommon	$> 25^o$	Deep
	Soil Falls	Bouncing, rolling, free fall	$M_L \geq 4.0$	Moderately Common	$> 40^o$	Shallow
	Disrupted Soil Slides	Translational sliding	$M_L \geq 4.0$	Very Abundant	$> 15^o$	Shallow
	Soil Avalanches	Complex, involving sliding, flow, and occasionally free fall	$M_L \geq 6.5$	Abundant	$> 25^o$	Shallow
Coherent Landslides	Rock Slumps	Rotational sliding	$M_L \geq 5.0$	Moderately Common	$> 15^o$	Deep
	Rock Block Slides	Translational sliding	$M_L \geq 5.0$	Uncommon	$> 15^o$	Deep
	Soil Slumps	Rotational sliding	$M_L \geq 4.5$	Abundant	$> 7^o$	Deep
	Soil Block Slides	Translational sliding	$M_L \geq 4.5$	Abundant	$> 5^o$	Deep
	Slow Earth Flows	Translational sliding and internal flow	$M_L \geq 5.0$	Uncommon	$> 10^o$	Variable
Lateral Spreads and Flows	Soil Lateral Spreads	Translation on fluid basal zone	$M_L \geq 5.0$	Abundant	$> 0.3^o$	Variable
	Rapid Soil Flows	Flow	$M_L \geq 5.0$	Moderately Common	$> 2.3^o$	Shallow
	Subaqueous Landslides	Generally lateral spreading or flow; occasionally sliding	$M_L \geq 5.0$	Uncommon	$> 0.5^o$	Variable

Shallow $\leq$3 m, Deep > 3m

Very abundant: > 100,000 – Abundant: 10,000 to 100,000 – Moderately Common: 1,000 to 10,000 – Uncommon: 100 to 1,000 in the 40 historical earthquakes

After investigating 40 historical earthquakes, Keefer (1984) stated that although all types of landslides posed some degree of hazard to human life, at least 90% of the casualties were caused by rockfalls, rock avalanches, and rapid soil flows. Even though the relative abundance of the rock avalanches and rapid soil flows are low (see Table 2.1), they are very dangerous because they can travel several kilometers at high velocities over gently sloping ground.

Table 2.2 Number of documented landslides produced by earthquakes (Keefer, 2002)

Earthquake Location	Year	Magnitude*	Number of Landslides	Approximate Percentage by Category		
				Disrupted (%)	Coherent (%)	Lateral Spreads and Flows (%)
Daly City, CA, USA	1957	5.3	23	48	30	22
Guatemala	1976	**7.5**	~50,000	Most	?	?
Mt. Diablo, CA, USA	1980	5.8	103	83	17	0
Mammoth Lakes, CA, USA	1980	**6.2**	5,253	>98	<1	<1
Coalinga, CA, USA	1983	**6.5**	9,389	>97	<2	<1
San Salvador, El Salvador	1986	**5.7**	>216	>93	<5	<2
Loma Prieta, CA, USA	1989	**6.9**	~1,500	74#	26#	0#
Northridge, CA, USA	1994	**6.7**	>11,000	>90	<9	<1
Hygoken-Nanbu, Japan	1995	**6.9**	674-747	81 to 83	13 to 15	3 to 4
Umria-Marche, Italy	1997	**6.0**	100-124	61	34	5
Chi-Chi, Taiwan	1999	**7.7**	22,000	>85	11 to 15	<4

* Magnitudes in bold are moment magnitudes; others are Richter surface-wave (Mt.Diablo) or Richter local (Daly City) magnitude # Percentages determined in central area only

2.2.1. Disrupted Landslides

Disrupted landslides include rock falls, disrupted rock slides, soil falls, rock avalanches, disrupted soil slides, and soil avalanches. Landslides in this category are usually initiated on steep slopes; they travel relatively rapidly and are able to transport the landslide material far beyond the bases of the steep slopes. The materials consist of boulders and masses of rock fragments, small blocks of soil that move downslope by falling, bouncing, and rolling (rock falls and soil falls). They also move downslope by translational sliding such as rock slides and disrupted soil slides, and complex mechanism involving both sliding and fluid like flow such as rock and soil avalanches (Keefer, 1984;

Keefer, 2002). With the exception of rock avalanches all types of slides in this category have initial failure depths of less than 3 m.

From the landslides identified in the southern Santa Cruz Mountains triggered by the 1989 Loma Prieta earthquake, 950 (74%) were disrupted landslides. The earthquake initiated these landslides throughout an area of about 15,000 km^2. These disrupted landslides mainly consisted of rock falls, rock slides, and disrupted soil slides. The deposits of these slides consisted of shallow highly disrupted masses of soil and rock. 33% of these disrupted landslides were adjacent to roads, and involved artificial cuts. 20% originated on low slopes, along streambanks, 42% in midslope localities, and 5% near ridgecrests (Keefer, 1998). Disrupted landslides in all different geological units were typically in materials that were weakly cemented, closely fractured, and intensely weathered. Most of the rockfalls, rock slides, and soil slides had volumes less than 100 m^3.

Similar to the 1989 Loma Prieta earthquake, the most common type of landslides triggered by the 1994 Northridge earthquake were highly disrupted, shallow falls and slides of debris, extending over the entire area of the landslide limit (Harp and Jibson, 1996). These landslides accounted for more than 90% of the total 11,000 landslides over an area of about 10,000 km^2. Landslides were several decimeters to a few meters deep and consisted of dry, highly disaggregated material moved down-slope to flat areas or near the base of the steep slopes. Average volumes of these landslides were less than 1000 m^3. The larger disrupted slides commonly had runout paths of more than 50m, and a few traveled as far as 200 m from the bases of steep slope that they were originated (Harp and Jibson, 1996)

1999 Chi-Chi earthquake triggered more than 20,000 landslides which more than 85% of them were disrupted landslides. More specifically, 63% of the slides were disrupted soil slides and falls and 22% were toppling and rock fall (Khazai and Sitar, 2004). The shallow landslides were not associated with a particular geologic unit. They

were as deep as the root zone of the vegetative cover from several decimeters to a meter deep.

Shallow landslides and rockfalls on steep natural slopes are usually the most common sight in mountainous terrains following a major earthquake. Shallow landslides often join together into sheets covering entire slopes, which makes it difficult to delineate individual landslide masses (see Figure 2.2). These types of landslides commonly strip off the vegetative cover of the slopes and leaving them highly vulnerable to aftershocks (Khazai, 2004).

2.2.2. Coherent Landslides

The second major category identified by Keefer (1984) is coherent landslides. This category includes rotational slides (rock slumps and soil slumps - Figure 2.3 (a)), translational slides (rock block slides and soil block slides - Figure 2.3 (b)), and slow earth flows, which move by a combination of translational sliding and flow (Keefer, 2002). The landslide material shows a slight to moderate internal disruption and typically moves relatively slowly. They commonly occur on moderately steep slopes and they are relatively thick with initial failure depths larger than 3m. Coherent landslides cause significant damage to houses, roads, and pipelines. The primary threat to pipelines comes from landslides where failure surfaces are several meters or greater in depth (Baum et al., 2008).

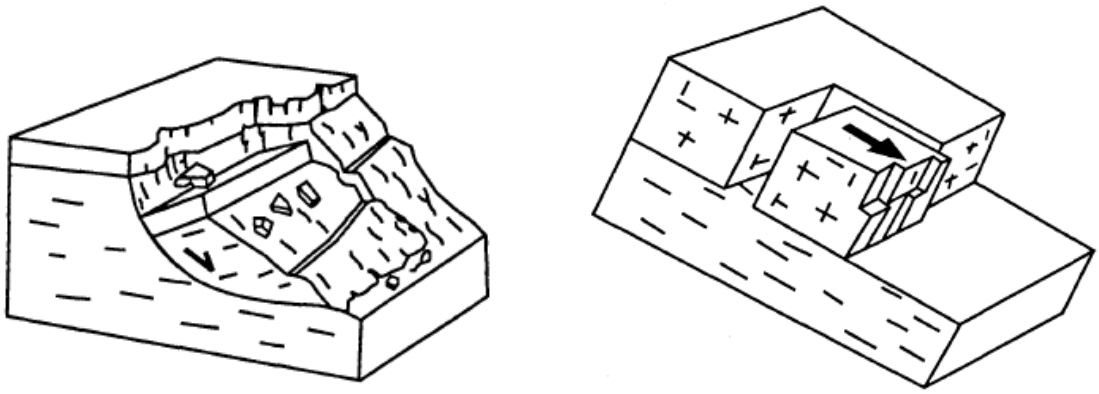
The 1989 Loma Prieta earthquake triggered a greater proportion of large deep-seated and coherent slides than Chi-Chi and Northridge earthquakes (Khazai and Sitar, 2004). In the southern Santa Cruz Mountains, 330 (26%) of the mapped landslides triggered by 1989 Loma Prieta earthquake were slumps or block slides. Commonly these landslides were bounded at their heads by distinct scarps or fissures and many contained internal fissures and compressional features. Figure 2.4 shows the nomenclature for the parts of an idealized deep-seated coherent landslide.

Image: Quickbird True Color Pan Sharpened Satellite Imagery
Acquistion Date: June 26, 2005
Spatial Resolution: 0.6 Meters

Image: Quickbird True Color Pan Sharpened Satellite Imagery
Acquistion Date: June 3, 2008
Spatial Resolution: 0.6 Meters

Figure 2.2 Pre and post earthquake satellite images showing disrupted landslides near epicenter in 2008 Wenchuan earthquake

The slumps and block slides ranged in size from a few meters wide to as much as 980 m long by 1,300 m wide for large landslides in the Summit Ridge area (Keefer, 1998). Drilling and other evidence showed that the large landslides in the summit area were at least 27 m, and possibly more than 100 m, deep. Measured displacement for coherent landslides ranged from less than 1 cm to more than 3 m. However, independent of the size of the landslides, most displacements were from 10 to 100 cm. 33% of these coherent landslides triggered in midslope localities, 30% involved roadcuts, fills or embankments, 22% were along ridge flanks and 15% originated on ridgecrests. The most common landslides under this category were small slumps in roadfill. Most of the coherent landslides of Loma Prieta earthquake were in unconsolidated Quaternary deposits (especially artificial fill, pre-existing landslide deposits, or alluvium) or in poorly to moderately indurated Tertiary sedimentary rocks. These deep-seated coherent landslides caused considerable damage to residences, roads, and other structures (Keefer, 1998).



a) b)

Figure 2.3 Diagrams showing idealized coherent landslides a) rock or soil slumps, consisting of single coherent block, showing movement with significant headward rotation on a curved basal shear surface b) rock or soil block slide, single coherent block showing translational movement on planar basal shear surface (After Keefer (1998))

Out of the 11,000 landslides triggered by the 1994 Northridge earthquake less than 9% were classified as coherent landslides. Although deep, coherent landslides were far less numerous than disrupted slides, they significantly added to the total volume of landslide material because of their size (see Figure 2.5). A few of these deep rotational slumps and block slides exceeded 100,000 m^3 in volume. The largest single landslide caused by the earthquake was a rotational slump/block slide which had a volume of 8 x 10^6 m^3. Slumps and block slides having volumes of several hundred to thousands of cubic meters occurred in some of the more competent sedimentary units in the Santa Susana Mountains and in the mountains north of the Santa Clara River. Most of these landslides were reactivations of previously existing features. Many pump pads, pipelines, and roads were undermined by deep slumps. Dozens of residences in the central and eastern Santa Monica Mountains were moderately or severely damaged by reactivation of old, deep block slides. Moreover, fill failures damaged hundreds of homes and other buildings (Harp and Jibson, 1996).

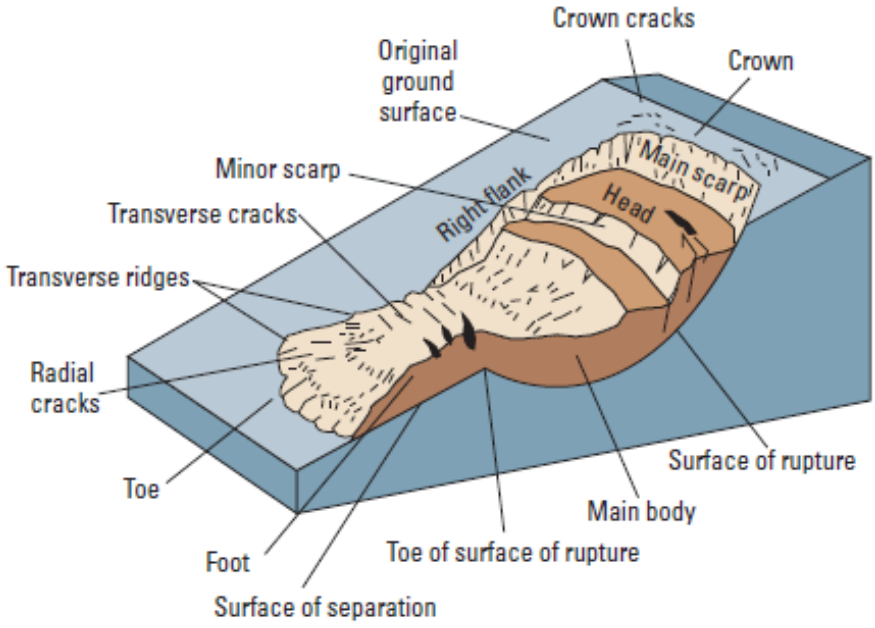


Figure 2.4 The nomenclature for the parts of an idealized deep-seated coherent landslide (Baum et al., 2008).

Deep-seated coherent landslides triggered by the 1999 Chi-Chi earthquake were far less numerous than shallow and disrupted slides and falls. Coherent deep seated landslides, which had volumes ranging up to tens of millions of cubic meters only accounted for 11% of failures (Khazai and Sitar, 2004). Even tough, they were less common, they were very destructive because even if the amount of displacement is relatively modest they involve relatively large amount of materials (Khazai, 2004). Two of these landslides were Tsao-Ling and Mt. Juo-Feng-Err landslides. They were truly catastrophic and were responsible for most of the loss of life caused by landslides (Khazai and Sitar, 2004). Several deep-seated landslides were triggered at the Chungliao substation in Central Taiwan. One of these landslides had an estimated volume of 1 million m^3. The maximum displacements at the crest of the landslide were only about 1.5 m horizontal and 2.1 m vertical. Because of this landslide the whole substation went out of service. This caused a major power blackout in northern Taiwan which was a major contributing source to overall cost (Khazai, 2004).

The 2004 Mid Niigta prefecture earthquake in Japan triggered more than one hundred deep-seated coherent landslides. Field investigations and the interpretation of aerial photographs taken before and after the event suggested that the most important factors for the deep-seated coherent landslides were the reactivation of pre-existing landslides and undercutting of slopes (Chigira and Yagi, 2006). A number of deep-seated coherent landslides which had volumes as large as several thousand cubic meters were also reported after the 2002 Avaj, Iran earthquake (Mahdavifar et al., 2006).

2.2.3. Lateral Spreads and Flows

The third major category identified by Keefer (1984) is lateral spreads and flows. This category includes the landslides for which fluid-like flow is the predominant movement mechanism. These landslides occur only in soil materials and involve blocks or relatively intact material moving on a subsurface liquefied zone (soil lateral spreads).

On the other hand rapid soil flows involve completely liquefied masses that move by fluid like flow. Most of the time, these landslides are the results of soil liquefaction in saturated sands, gravels, and silts; occasionally they result from seismic disturbance in sensitive clays. Lateral spreads and flows usually occur in clayey soils; volcanic soils; talus and old landslide deposits; and alluvial, deltaic, lacustrine and coastal deposits. Landslides in this category usually start and move on gentle to nearly level slopes, move fast, and can transport material large distances (Keefer, 2002).

Lateral spreads and flows comprised about 6% of the total number of landslides reported from 40 earthquakes by Keefer (1984). One well-developed liquefaction-induced lateral spread was mapped in Tapo Canyon, in 1994 Northridge earthquake. Keefer (1998) didn't report any lateral spreads and flows for Loma Prieta earthquake. Lateral spreads and flows reported for the 1999 Chi-Chi earthquake was less than 4% of the total landslides (Keefer, 2002).

Figure 2.5 Loma Verde deep-seated landslide triggered by 1994 Northridge earthquake, Light-colored, arcuate scarps visible in upper center of photo each had 1-2 m of displacement (Harp and Jibson, 1996).

2.3. Seismic Response and Deformation of Slopes

There are usually three main categories for the analysis of slopes under seismic loading conditions. The pseudo-static analysis and the Newmark analysis have been commonly used in practical applications and are relatively simplistic. The most sophisticated method for seismic slope stability calculations is known as a dynamic analysis or a stress-deformation analysis (Kramer, 1996) and it typically incorporates a finite element/difference mathematical model. In this type of analysis, ground motion is incorporated in the form of an acceleration time history. Seismically induced permanent strains in each element of the finite element/difference mesh are integrated to obtain the permanent deformations (Blake et al., 2002).

2.3.1. Pseudostatic Slope Stability Analysis

Since the early 20th Century, engineers have used pseudostatic analysis to determine the seismic stability of slopes. This approach applies the earthquake load as an additional static force on the slope; therefore it neglects the cyclic nature of the earthquake loading. Figure 2.6 shows a simple slope failure which has a planar slip surface with an angle of α from horizontal. The magnitudes of the pseudostatic forces are given by:

$$F_h = ma_h = \frac{a_h W}{g} = k_h \mathrm{W} \qquad (\mathbf{2.1})$$

$$F_v = ma_v = \frac{a_v W}{g} = k_v W \qquad (\mathbf{2.2})$$

F_h and F_v are the horizontal and vertical pseudostatic forces that stem from the earthquake loading and act through the centroid of the failure mass , m is the total mass (W is the total weight) of the sliding materials (sliding block over line AB in Figure 2.6), a_h and a_v are the horizontal and vertical earthquake accelerations, g is the gravitational

constant, $k_h = a_h/g$ and, $k_v = a_v/g$ are dimensionless horizontal and vertical pseudostatic seismic coefficients. However, in standard pseudostatic analysis, vertical earthquake forces are usually ignored because they have much less effect on the stability of slopes than horizontal forces.

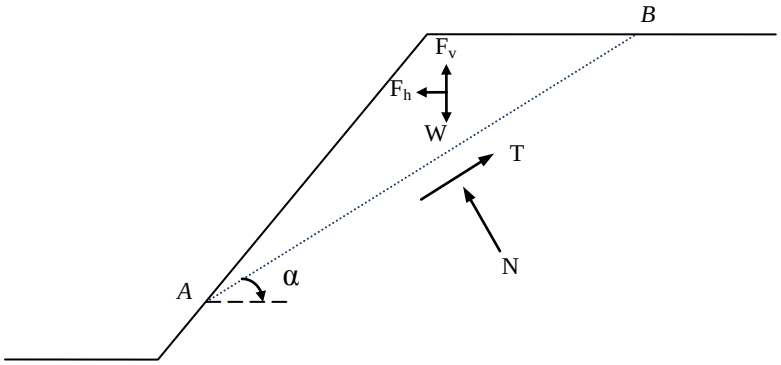


Figure 2.6 Planar slope failure with pseudostatic forces acting on the slope

Basically, pseudostatic analysis is an evaluation of the limit equilibrium method considering the inertial effects of the earthquake. The result of the pseudostatic analysis is a factor of safety against failure wherein if it is 1.0 or greater it means the slope is stable under the earthquake load. Ignoring the vertical forces the factor of safety of slope under earthquake loads can be written as:

$$FS = \frac{resisting\ forces}{driving\ forces} = \frac{cL + (Wcos\alpha - F_h sin\alpha)tan\phi}{Wsin\alpha + F_h cos\alpha} \qquad (\mathbf{2.3})$$

where c and ϕ are the Mohr-Coulomb strength parameters that describe the cohesive and frictional strength of the soil on the failure plane and L is the length of the failure plane.

The pseudostatic analysis is computationally very simple and easy for integration with GIS, however it has serious limitations (Kramer, 1996). The method is inherently conservative because the cyclic force is replaced by a constant force equal to the

maximum transient force. A slope may remain intact during an earthquake even though the maximum acceleration passes the critical acceleration, however with the pseudostatic method if the critical acceleration is exceeded the factor of safety is less than zero. The length of time or the number of instances that the maximum acceleration is greater than the critical acceleration does not affect the factor of safety.

2.3.2. Newmark Displacement Method

The factor of safety calculated by the pseudostatic analysis provides an index of stability for the slope, however it doesn't present any information about the deformation of the slope after the critical acceleration value is exceeded. Newmark (1965) proposed a model that calculates the cumulative displacement of a slope as a result of the earthquake loading. Newmark's original intent was to use the model for estimating the seismic performance of dams and embankments (Newmark, 1965). However, Newmark's method also has been successfully applied to landslides in natural slopes (Jibson, 1993; Miles and Ho, 1999). In the Newmark displacement method, the landslide is modeled as a rigid-plastic frictional block resting on an inclined plane (see Figure 2.7). The rigid block has a critical acceleration value (a_c) which is defined as the minimum base acceleration required to overcome shear resistance and initiate sliding. The block doesn't deform internally and deforms plastically along a discrete basal shear surface when the critical acceleration is exceeded. In the Newmark displacement method, an acceleration-time history is applied to the rigid block with a known critical acceleration value. Accelerations higher than the critical acceleration value of the slope cause the slope to move. Accelerations below this value don't cause any displacement. The portions of the records that exceed the critical acceleration value are integrated and the velocity profile is obtained. After integrating the velocity profile, the displacement history of the block is obtained (see Figure 2.8). Newmark displacement is a useful index of how a slope is

likely to perform during an earthquake (Jibson et al., 2000). The main assumptions of the Newmark displacement methods can be summarized as:

- The soil behaves in a rigid, perfectly plastic manner.
- Displacements occur along a single, well defined slip surface.
- Soil does not undergo strength loss as a result of shaking.

In order to simplify Newmark analysis, the following additional limiting assumptions are usually imposed (Jibson, 1993):

- Displacement occurs only in downslope direction and vertical accelerations may be ignored.
- The static and dynamic shearing resistances of the soil are taken to be the same.
- The effects of pore pressure are neglected.
- The critical acceleration is not strain dependent and thus remains constant throughout the analysis.

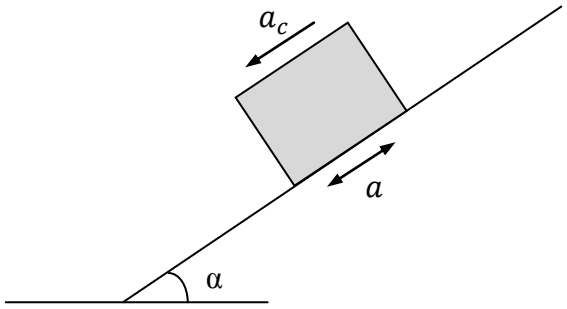


Figure 2.7 Sliding-block model used for Newmark analysis

Newmark (1965) showed that the critical acceleration (a_c) of a potential landslide block is a simple function of the static factor of safety and the landslide geometry:

$$a_c = (FS - 1)g \sin \alpha \qquad (\mathbf{2.4})$$

where g is the gravitational acceleration and α is the angle from the horizontal that the center of mass of the potential landslide block first moves and it can be approximated as the slope angle (Jibson et al., 2000).

Newmark's sliding block analogy has been extended to account for the deformable response of the sliding body (Makdisi and Seed, 1978). Seed and Martin (1966) introduced a technique for estimating the seismic loading on a sliding mass which included the deformable response of the sliding body. A dynamic analysis is performed to calculate the acceleration and shear stress distribution within the slope and the seismic load is characterized by the horizontal equivalent acceleration (HEA), which is the average acceleration within the slide mass (Seed and Martin, 1966). Lin and Whitman (1983) studied the decoupled approximation using a lumped mass, shear beam model with linear elastic material properties. Systems were analyzed for shallow, deep, and intermediate sliding (Lin and Whitman, 1983; Rathje and Bray, 1999).

Many studies have been carried out to develop simplified approaches to obtain Newmark displacements without directly using acceleration-time histories. Empirical relationships are developed using real earthquake records to estimate displacements as a function of shaking intensity and critical acceleration (Ambraseys and Menu, 1988; Jibson, 1993; Jibson, 2007; Saygili and Rathje, 2008).

2.3.3. Finite Element/Difference Methods

Finite element and finite difference elasto-plastic analysis of geotechnical problems has been used for large scale and research projects, but has not been used as much for routine applications. However, as more emphasis is placed on making a reliable prediction of permanent deformations of earth structures, practical applications with nonlinear numerical codes have increased (Itasca, 2008). In these methods, materials are represented by elements, or zones, which form a grid that is adjusted by the user to fit the

shape of the slope to be modeled. Each element behaves according to a prescribed linear or nonlinear stress/strain law in response to the applied forces or boundary restraints. In finite element/difference methods, no prior assumption needs to be made about the shape or location of the failure surface. Failure occurs through the zones within the soil mass where the shear strength is unable to sustain the applied shear stresses. Finite element/difference methods illustrate progressive failure up to and including overall shear failure. By contouring shear strains in the zones, it is possible to highlight failure surfaces (Blake et al., 2002). Irreversible displacements and other permanent changes are modeled automatically (Itasca, 2008).

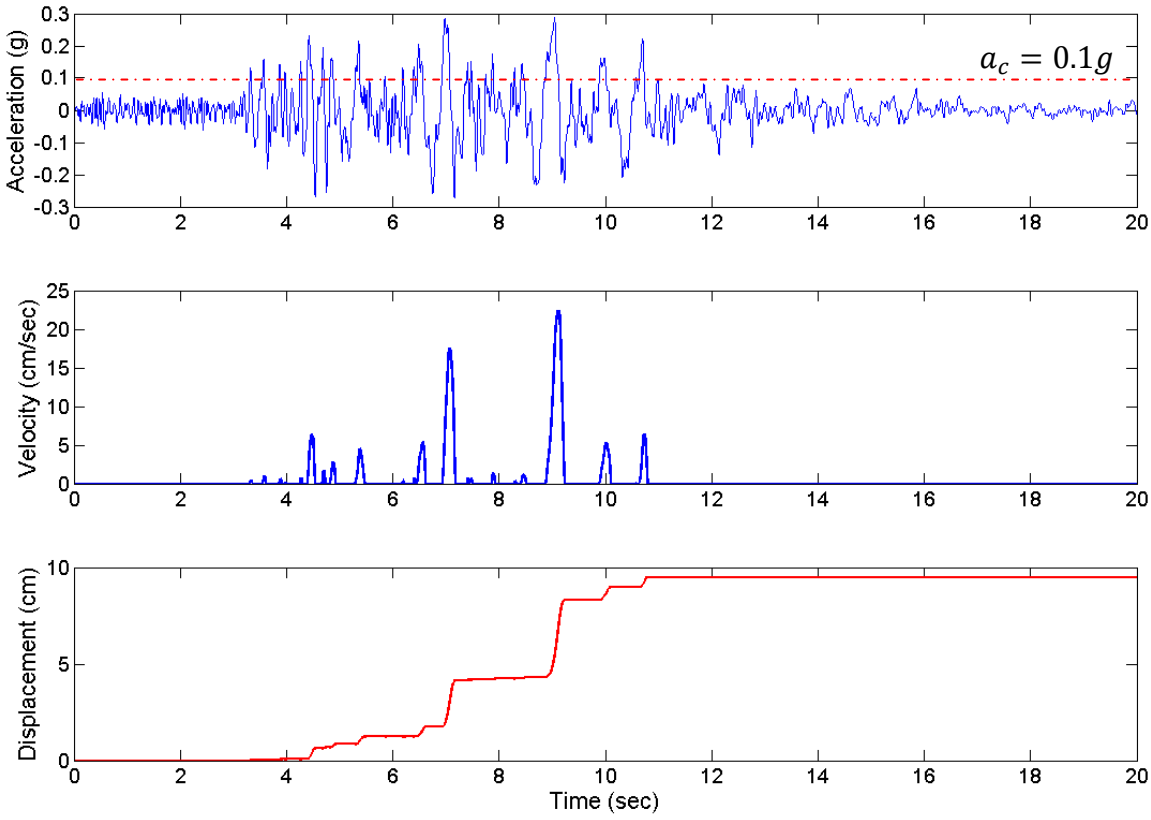


Figure 2.8 Demonstration of the Newmark method double integration

2.4. Application of GIS to Regional Landslide Hazard Zonation

Geographic Information Systems (GIS) provides an excellent framework for efficiently storing, updating and analyzing the data required for regional seismic slope stability modeling. In the last three decades, the use of GIS for landslide hazard mapping has increased rapidly and different approaches have been developed. Because the form of the land surface has a essential function in landslide mechanisms, much of the increase in use of GIS has been driven by the increasing availability of Digital Elevation Models (DEM) (Carrara and Pike, 2008). GIS methods can be grouped into two main categories: qualitative and quantitative methods (Guzzetti et al., 1999).

Qualitative methods are subjective and define the hazard in descriptive terms. Qualitative methods are based on the judgment of person performing the hazard assessment without expressing a precise value or quantification for it. Qualitative methods are usually based on field geomorphological hazard and overlay a combination of index maps or parameter maps with weights (Khazai, 2004). Geomorphic analysis combines directly mapped landslides with their geomorphic settings to generate hazard maps. It is based on the direct relationship between the occurrence of slope failure and the causative terrain parameters. Causative factors are ranked and weighted according to their assumed or expected importance in causing slope failure (Guzzetti et al., 1999; Luo, 2006).

On the other hand, quantitative methods are based on the quantification of hazard either using process-based or statistical approaches. Quantitative methods produce numerical estimates of the occurrence of landslide in any hazard zone (Guzzetti et al., 1999; Khazai, 2004).

The statistical approach observes the relationship between each factor and the past landslide distribution. In statistical models, the aim is to correlate and compare parameters that are potentially significant for the initiation of landslides. Statistical, “black-box” approaches are based on the analysis of the functional relationships between

instability factors and the past distribution of landslides (Guzzetti et al., 1999). The basic assumption of the statistical approach is that the factors which caused landslides in the past will generate landslides in the future (Luo, 2006).

Process-based models rely upon the mechanics of the failure and employ models of seismic slope stability to assess the deformation of the slope following an earthquake. The infinite slope stability method coupled with Newmark's displacement method is the most popular process-based model for assessing the earth-quake induced landslide susceptibility on a regional scale (Wieczorek et al., 1985; Miles and Ho, 1999; Jibson et al., 2000).

The infinite slope model is a one dimensional process-based model which describes the stability of slopes with an infinitely long failure plane. The assumption of infinite slope is valid if (1) the landslide mass is thin compared to its length (2) the failure surface is parallel to the ground surface and (3) failure occurs as basal sliding (Miles and Ho, 1999). Figure 2.9 shows the slope geometry to be used for an infinite slope model. The static factor of safety for the infinite slope model can be written as:

$$FS = \frac{c'}{\gamma d \sin\alpha} + \frac{\tan\phi'}{\tan\alpha} - \frac{m\gamma_w \tan\phi'}{\gamma \tan\alpha} \qquad (\mathbf{2.5})$$

where ϕ' is the effective friction angle, c' is the effective cohesion, α is the slope angle, γ is the material unit weight, γ_w is the unit weight of water, d is the slope-normal thickness of the failure slab, and m is the proportion of the slab thickness that is saturated (Jibson et al., 2000). The infinite slope model which consists of a simple equation, is the most popular slope stability technique used with GIS (Miles and Ho, 1999). However, landslide hazard analyses using the infinite slope model are applicable only to shallow translational landslides. The infinite slope model gives reliable results in the case of surficial landslides with depth-length ratios smaller than 0.1 (Xie et al., 2003). On the

other hand, the infinite slope model cannot satisfactorily analyze deep-seated, rotational failure (Luo, 2006; Baum et al., 2008).

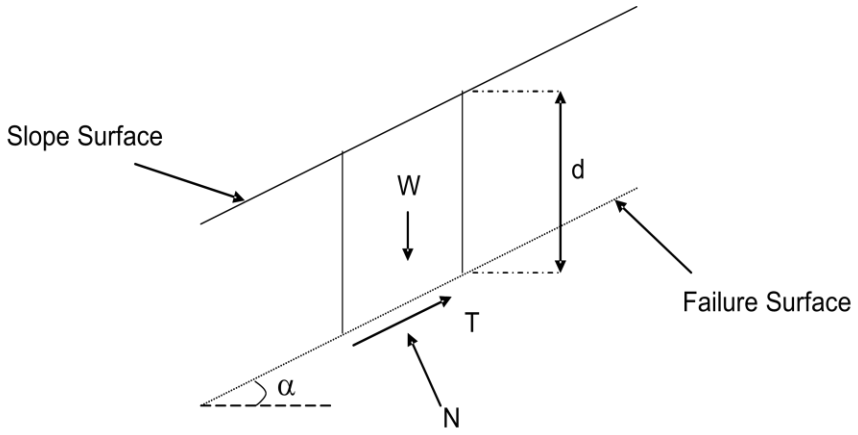


Figure 2.9 Infinite Slope Model

A flow chart showing the basic steps in using the infinite slope model in conjunction with Newmark's displacement method can be seen in the Figure 2.10. The mapping unit is a grid-cell. The analysis is performed for each grid-cell. First, using compiled shear-strength data from literature or reports, each unit on the geologic map is assigned effective friction (ϕ') and effective cohesion (c') values. Next, a slope map is created from the digital elevation model. After that, using the shear strength and slope data in the factor of safety equation (Eq. 2.5) for the infinite slope model, the factor of safety for each cell is calculated. Slab thickness and degree of saturation also need to be assigned. Subsequently, the critical acceleration value is calculated using the slope angle and the factory of safety in Equation 2.4. Finally the Newmark displacements can be calculated in two ways. The first way requires an acceleration-time history. For each grid, using the critical acceleration value and the acceleration-time history, Newmark displacement is calculated with double integration (Miles and Ho, 1999). The second way

is to use a simplified method without using acceleration-time histories (Jibson et al., 2000; Saygili and Rathje, 2008).

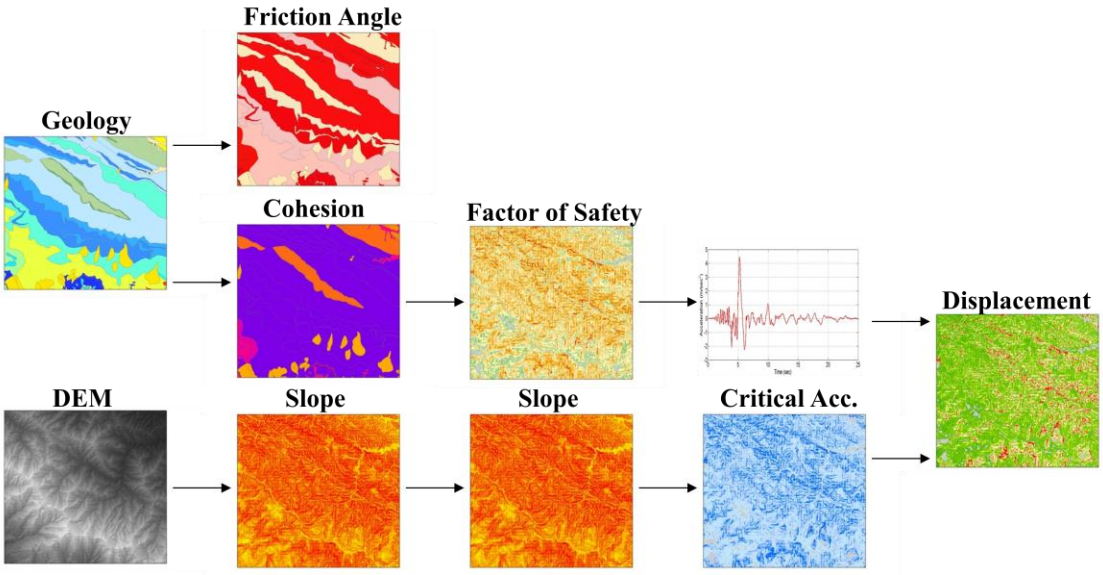


Figure 2.10 Flowchart for regional seismic hazard analysis using infinite slope model with Newmark displacement analysis

As stated earlier, the infinite slope models are only suitable for shallow surficial landslides (Mankelow, 1997; Xie et al., 2003; Luo, 2006; Baum et al., 2008). Therefore, there is a need for coupling seismic slope stability analysis for deep-seated coherent landslides with GIS techniques to evaluate the slope stability on a regional scale. Some researchers used conventional two-dimensional 2-D moment equilibrium analyses coupled with ground-water models to estimate static factor of safety using GIS (Miller, 1995). Also some scientists used 3-D analysis to calculate the static factor of safety of slopes in GIS environment (Xie et al., 2003). Xie et al. (2003) used a column-based three-dimensional limit equilibrium method to calculate static safety factors. Some researchers used 2D slope profiles with different limit equilibrium models and Newmark's method to analyze deep-seated landslides triggered by the 2001 El Salvador earthquake (Luo, 2006).

2.5. Summary

Earthquake triggered landslides pose a significant threat to both life and property, and adversely affect the economic and social life of the hazard region. Predicting where and in what shaking conditions earthquakes are likely to trigger landslides is an important element in regional seismic hazard assessment (Jibson et al., 2000).

Earthquake triggered landslides are usually categorized into three main categories as, disrupted landslides, coherent landslides, and lateral spreads and flows (Keefer, 1984). Disrupted landslides are the most common type of landslides worldwide. Deep-seated coherent landslides are usually far less numerous than disrupted slides. However, they are very destructive because even if the amount of displacement is relatively modest they involve relatively large amount of materials (Khazai and Sitar, 2004).

There are three main categories for the analysis of slopes under seismic loading conditions: the pseudo-static analysis, the Newmark analysis and stress-deformation analysis. GIS methods for regional earthquake triggered landslide hazard analysis can be grouped into two main categories: qualitative and quantitative methods. Qualitative methods are subjective and define the hazard in descriptive terms. Quantitative methods produce numerical estimates of the occurrence of landslide in any hazard zone.

Infinite slope model in conjunction with Newmark's displacement method has been commonly used for regional landslide hazard analysis. However, the infinite slope models are applicable only to shallow translational landslides. There is need for research for seismic slope stability analysis of deep-seated coherent landslides on a regional scale.

CHAPTER 3

SOFT COMPUTING METHODS

3.1. Introduction

During the recent decades there has been an enormous research and development activity in soft computing (SC) methods. Building metamodels (or surrogate models) based on the data obtained from computer experiments and using computationally inexpensive predictions based on these metamodels have been widely used in engineering applications due to the significant computational cost of using high-fidelity simulations.

Running complex computer simulations is an important part of today's engineering analysis. Computer simulations using nonlinear finite element method (FEM) or finite difference method (FDM) may be powerful tools when a detailed damage estimate is required for an individual geostructure or a small number of geostructures. However, despite the high computing power of today's hardware, single evaluations of FEM and FDM analyses can take minutes to hours, if not longer. Therefore, in the case of an assessment involving many geostructures with different design variables these powerful tools become computationally expensive and inefficient. One solution to this problem is using approximate simulations or simpler mathematical models where accuracy is sacrificed to decrease computational time. Using a one dimensional infinite slope model instead of a two or three dimensional nonlinear finite element/displacement model for a regional landslide hazard analysis can be considered as an example of this simplification. An alternative solution to this problem is to create metamodels to substitute individual simulations. In this solution, design variables are carefully chosen using a design of experiments (DOE) methodology to cover a predetermined range of values and computer experiments are performed at these chosen points. The design variables and the responses from the computer simulations are then combined to

construct functional relationships (metamodels) between the inputs and the outputs. Methods for creating metamodels include response surface methodology, kriging, inductive learning, artificial neural networks and support vector machines.

Computer simulations consist of supplying a vector of design variables (inputs) $\boldsymbol{x}$ into a computer code and calculating a vector of responses (outputs) $\boldsymbol{y}$. The true function of the computer simulation is

$$y = f(x) \tag{3.1}$$

where f is a measurable function but expensive to compute and $x \in T$ where T is a defined domain or region of interest for the input space. The purpose of metamodeling is to compute function f at a set of points $x_1, \ldots, x_n$ which are chosen using the design of experiments methodology and construct an approximate function

$$\hat{y} = g(x) \tag{3.2}$$

where $\hat{y}$ is the predicted response and function g is the metamodel. The relationship between x and y then becomes

$$y = g(x) + \varepsilon \tag{3.3}$$

where ε represents both the error of approximation and measurement (random) errors. However, the measurement errors do not exist in the case of computer experiments because unlike the physical experiments, the computer experiments are deterministic which means responses from running the simulations with the same design variables will be equal.

In general, constructing metamodels consists of four main steps: 1) choosing an appropriate experimental design for creating input parameters for the computer simulations (Design of Experiments), 2) running computer simulations for the selected

design points and calculating the responses, 3) choosing a model to represent the observed data and fitting the model to the data, 4) validating the model. A flowchart of constructing metamodels can be seen in Figure 3.1. In the sequential metamodeling approach which is adopted in this study, experimental design starts with a certain number of computer experiments and if the predictive accuracy of the metamodel is not satisfactory, more experimental runs are added until there is no further improvement in the prediction accuracy.

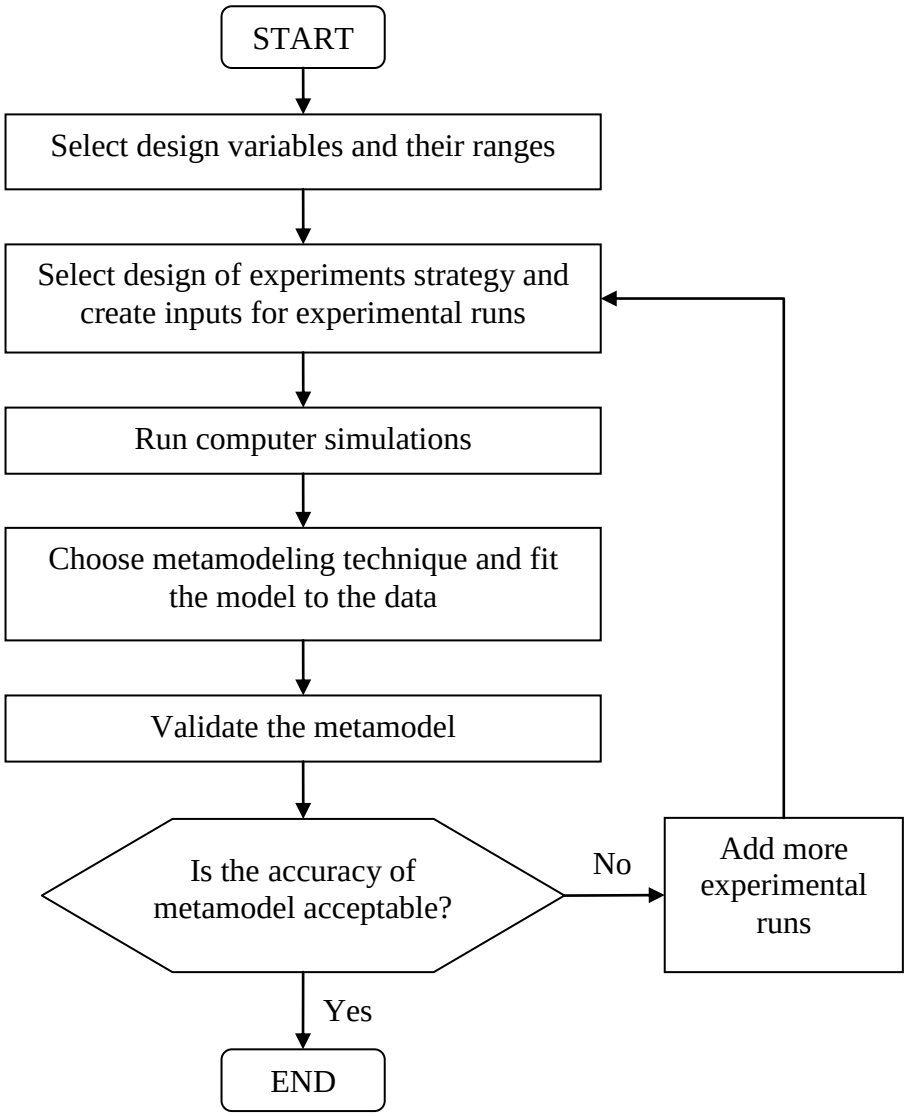


Figure 3.1 Flowchart for constructing metamodels

One of the most popular metamodeling techniques has been the response surface methodology (RSM). RSM metamodels are usually built using central composite design of experiments and they typically employ lower order polynomial models fit using least square regression techniques (Myers and Montgomery, 1995). RSM was primarily intended for physical experiments with measurement error but it has successfully been used in many engineering applications involving computer experiments. The reasons for the popularity of RSM are its simplicity, maturity and the availability of software tools. However, as the number of design variables (inputs) increases, RSM loses its appeal because the number of experiments required to fit the polynomial becomes unmanageable. Moreover, RSM has limited capabilities to model non-linear responses. Theoretically, higher-order polynomials could be used to model a non-linear relation but instabilities may arise and again the number of experiments required to estimate all of the coefficients in the polynomial equation becomes too high (Simpson, 1998).

For highly non-linear systems and/or systems with high number of design variables, the use of machine learning techniques such as artificial neural networks, fuzzy systems and support vector machines has become popular due to the limitations of response surface methodology. The excellent performance of machine learning techniques in the modeling of non-linear relationships of multivariate dynamic systems also makes them popular tools for earthquake engineering applications (Lagaros and Tsompanakis, 2007).

Artificial neural networks (ANN), also called neural networks (NN), are one of the most popular machine learning techniques inspired by the architecture and behavior of biological neurons. ANN architectures are formed by three or more layers, including an input layer, an output layer and a number of hidden layers. Layers consist of numerous processing units (neurons) which are connected to each other with modifiable weights. These modifiable weights determine the nature of interaction between the interconnected neurons. ANN basically learns through the adaptation of interconnection

weights between the neurons. ANN has been successfully used in pattern recognition (classification), regression and optimization problems for many engineering applications.

A relatively new promising method for constructing metamodels is the support vector machines (SVM). SVM originated from the statistical learning theory developed by Vapnik and his collaborators (Vapnik, 1995; Vapnik, 1998). SVM can be considered as a new method for training polynomial models, neural networks, fuzzy models, or radial basis function classifiers/regressors (Kecman, 2001). There is a wide variety of problems that can be solved using SVM, however all the problems fit in two major groups: pattern recognition (classification) and functional approximation (regression).

3.2. Design of Experiments

The first step in building metamodels is the experimental design. Other than the choice of metamodeling method, the accuracy and efficiency of the metamodels are determined by the experimental design used to select the input design variables. Properly designed experiments are crucial for effective metamodel building.

An experimental design corresponds to a series of experiments to be conducted, expressed in terms of design variables set at predefined values. An experimental design is characterized by a matrix where each rows represents a single experimental run and the columns represent the values of the design variables.

Physical experiments have long been used to analyze the relationship between the input design variables and the response output. Measurement (random) error is inherent in physical experiments. Therefore, statisticians have developed a variety of techniques to design and analyze physical experiments that account for random variability by taking multiple observations with same design variables (replication), blocking, and randomization. Common choices of design for physical experiments include factorial, orthogonal, randomized block and optimal designs (Lam, 2008). Both physical and computer experiments are conducted to obtain data for modeling the relationship between

the input design variables and the response output, however because of the deterministic nature of the computer experiments an appropriate set of experimental design strategies has to be developed.

Computer experiments are different from physical experiments because repeated computer experiments with the same set of design variables give identical responses. Therefore, methods for controlling bias and noise in physical experiments such as replication, blocking and randomization are irrelevant for computer experiments. Metamodels seek to determine the input-output relationship using an approximate model with limited number of experiments. The reason uncertainty arises in metamodeling is that even though the response can exactly be calculated at any given input point, the exact functional form between the inputs and the outputs is not known. Any metamodel describing the input-output relationship is only an approximation. The difference between the actual response from the computer simulation and the response predicted from the metamodel is the approximation error. The main objective of experimental design for computer experiments is to select input design variables that will minimize the approximation error of the metamodel as well as the number of computationally expensive simulations. Two principles of experimental design for computer experiments can be stated as follows (Santner et al., 2003):

1. Due to the absence of measurement (random) errors, experimental designs should not take more than one observation at any set of input design variables.
2. Experimental designs should be space-filling so that they would provide information about all portions of the experimental region.

Intuitively, space filling designs are appealing for design of computer experiments because the design variables are spread out over the entire range of input space to increase the prediction accuracy of the metamodel. Many researchers advocate the use of space filling designs for computer experiments because they treat all regions of the input

design space equally (Simpson et al., 2001). Latin hypercube sampling is the most common space filling design used for computer experiments. Latin hypercube designs can be broadly categorized into two classes: randomly generated Latin hypercube designs and hybrid Latin hypercube designs. See Santner et al. (2003) for through discussions of space filling designs for computer experiments.

3.2.1. Randomly Generated Latin Hypercube Designs

Latin hypercube design (LHD) was first proposed by Mckay et al. (1979) for design of computer experiments. It was considered as an alternative to simple random sampling and stratified random sampling (Mckay et al., 1979). LHDs are computationally easy to construct and generally covers the design space well without replications (Park, 1994).

In order to obtain a LHD of size $\boldsymbol{n}$, the range of each design variable is divided into $\boldsymbol{n}$ intervals. This creates a total of $\boldsymbol{n^d}$ equal spaced partitions (cells) on the $\boldsymbol{d}$-dimension input space where $\boldsymbol{d}$ is the number of design variables. A set of $\boldsymbol{n}$ cells is chosen from the $\boldsymbol{n^d}$ population of cells by random permutation in such a way that the projections of the centers of each of the cells onto each axis are uniformly spread across the axis (Santner et al., 2003). Each permutation leads to a different LHD, for any given number of design variables and experiments there will be more than one possible LHD. For $\boldsymbol{n}$ experiments and $\boldsymbol{d}$ design variables there are $(\boldsymbol{n}!)^{\boldsymbol{d}}$ different LHDs. Considering a two dimensional input space (two design variables), a square grid with input points is a Latin square only if there is one point in each row and column. Two examples of LHD of size 5 for two design variables **A** and **B** can be seen Figure 3.2 (a) and (b). Not all LHDs are space filling, as it can be seen in Figure 3.2 (b), a Latin hypercube sample may have all its input points along one of the main diagonals, in this case even though the design has Latin characteristics it is not space filling. Figure 3.2 (b) is not a good design because two design variables are perfectly correlated therefore the effects of the two factors on the

output cannot be distinguished. Moreover, there is a large area in the input design space that is not explored. If a design like this is used to create a metamodel, the prediction accuracy of the metamodel will be poor in those unexplored areas. A randomly generated LHD can be quite structured such that the design variables can be highly correlated or it can lack good space filling properties. There are methods to find better LHDs by minimizing the pairwise correlations or maximizing the distance between the points (Joseph and Hung, 2008).

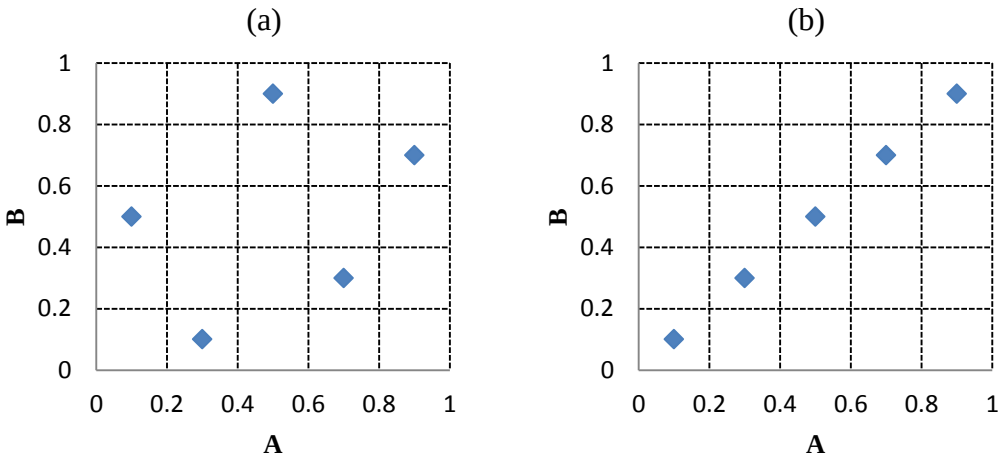


Figure 3.2 (a) A space filling Latin hypercube design (b) A non-space filling highly correlated Latin hypercube design

3.2.2. Hybrid Latin Hypercube Designs

In order to improve the space filling characteristics of Latin hypercube designs, other than using random permutation to choose the points, the assignments of the points should be refined by reference to an optimality criterion. The objective is to create a design by optimizing a criterion that describes a desirable characteristic of the design.

Some researchers proposed to find good LHDs by minimizing the correlations among design variables (Owen, 1994; Tang, 1998). Other researchers proposed maximizing the minimum Euclidean distance between points in the input space to find

the best LHD and called this design *maximin* LHD (Morris and Mitchell, 1995). Park (1994) proposed an optimal LHD by using the integrated mean squared criterion. Due to the highly combinational nature of selecting points in the design input space, finding the optimal LHD for high number of design variables is a very difficult task. Several algorithms using an exchange method for searching in the design space are proposed to find the optimal LHDs (Joseph and Hung, 2008).

A variety of software can be used to create hybrid Latin hypercube designs. JMP® by SAS Institute can be used to create LHDs with optimal spacing criterion or integrated mean squared error criterion. "Latin Hypercube Samples – lhs" package (Carnell, 2009) written in **R** statistical language can be used for a range of different designs including maximin LHDs, genetic LHDs and augmented LHDs.

From the designs that can be created using "Latin Hypercube Samples – lhs" package, augmented LHD is very useful for the sequential metamodeling approach followed in this study. It augments an existing Latin hypercube sample while maintaining the Latin properties of the design. Therefore it allows more points to be added into the original design if the predictive accuracy of the created metamodel is not satisfactory. A maximin LHD of size 10 with two design variables **A** and **B** is seen in Figure 3.3 (a). As it can be seen in Figure 3.3 (b), using the augmented LHD, 10 more points are added to the original design without disturbing the Latin properties of the sample. This method of adding points to the experimental design can be used until a desired predictive accuracy of the metamodel is achieved or until the predictive accuracy does not improve any further.

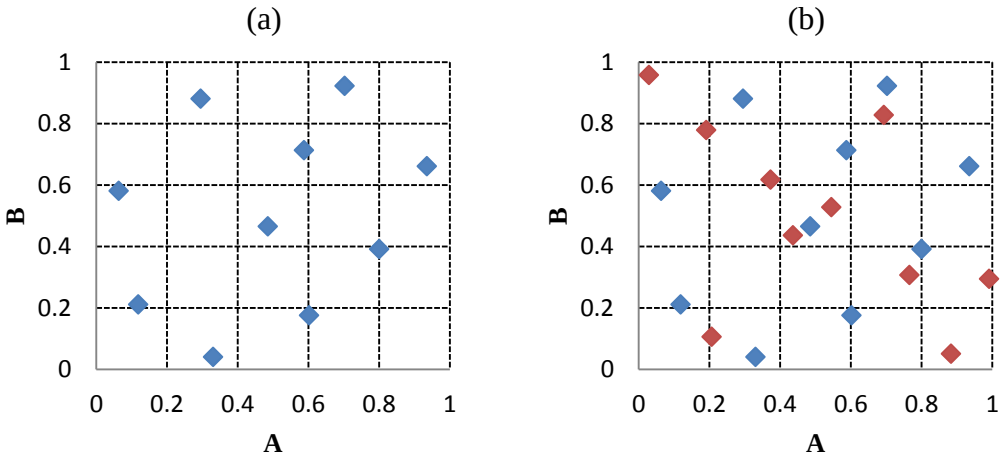


Figure 3.3. (a) Original maximin Latin hypercube design with 10 points (b) Augmented Latin hypercube design with additional 10 points

3.3. Metamodeling Techniques

After selecting an appropriate experimental design and running the computer simulations with the input parameters created by the design, the next step is fitting a suitable metamodel for the vector of inputs (x) and corresponding vector of responses (y). There are many methods for fitting the responses to the inputs and creating an approximation function. These methods include a wide variety of mathematical models from relatively simple regression analyses to more sophisticated machine learning algorithms.

3.3.1 Response Surface Methodology (RSM)

Response surface methodology investigates the functional relationships between several input variables and one or more response variables. The method was first introduced by Box and Wilson (1951) to determine the optimal conditions in the chemical investigations. A response surface equation can be simply defined as a polynomial regression to an input-output dataset. The method does not only refer to the

use of a response surface as a multivariate function to estimate a relationship but also to the process for determining the polynomial coefficients (Towashiraporn, 2004).

A quadratic response function is commonly used for RSM because it takes into consideration individual parameter effects, second order curvature or nonlinearity, and two parameter interactions. The quadratic response function is as follows:

$$\mathbf{y = \beta_0 + \sum_{i=1}^{d} \beta_i x_i + \sum_{i=1}^{d} \beta_{ii} x_i^2 + \sum_{i=1}^{d-1} \sum_{j>i}^{d} \beta_{ij} x_i x_j + \varepsilon} \qquad (\mathbf{3.4})$$

where

y = Response from the computer simulations

$\mathbf{x_i x_j}$ = Input design variables

$\boldsymbol{\beta_0, \beta_i, \beta_{ii}, \beta_{ij}}$ = Polynomial coefficients

$\boldsymbol{\varepsilon}$ = Error of approximation

d = Number of input design variables

Despite the second order terms in the quadratic equation, it can still be treated as a linear regression problem if the equation is written in the form of a general linear model as follows:

$$\mathbf{y = \beta_0 + \sum_{i=1}^{k-1} \beta_i z_i + \varepsilon} \qquad (\mathbf{3.5})$$

where k is the number of coefficients to be calculated and z is a vector of dummy first order variables that replaces the second order terms. Then a general matrix form of the linear regression models can be written as follows:

$$\mathbf{Y = Z\beta + \varepsilon} \qquad (\mathbf{3.6})$$

where Y is a vector of responses for n experiments,

$$\mathbf{Y} = \begin{bmatrix} \mathbf{y_1} \\ \mathbf{y_2} \\ \vdots \\ \mathbf{y_n} \end{bmatrix}_{\mathbf{nx1}} \tag{3.7}$$

Z is the matrix of dummy variables including first and second order design variables,

$$\mathbf{Z} = \begin{bmatrix} \mathbf{1} & \mathbf{z_{11}} & \mathbf{z_{12}} & \cdots & \mathbf{z_{1,k-1}} \\ \mathbf{1} & \mathbf{z_{21}} & \mathbf{z_{22}} & \cdots & \mathbf{z_{2,k-1}} \\ \vdots & \vdots & \vdots & \ddots & \mathbf{z_{11}} \\ \mathbf{1} & \mathbf{z_{n1}} & \mathbf{z_{n2}} & \cdots & \mathbf{z_{n,k-1}} \end{bmatrix}_{\mathbf{nxk}} \tag{3.8}$$

$\boldsymbol{\beta}$ is a vector of unknown coefficients,

$$\boldsymbol{\beta} = \begin{bmatrix} \boldsymbol{\beta_0} \\ \boldsymbol{\beta_1} \\ \vdots \\ \boldsymbol{\beta_{k-1}} \end{bmatrix}_{\mathbf{kx1}} \tag{3.9}$$

and $\boldsymbol{\varepsilon}$ is a vector of error terms with expectation $\mathbf{E}[\boldsymbol{\varepsilon}] = \mathbf{0}$. Therefore, a random vector Y has expectation of $\mathbf{E}[\mathbf{Y}] = \mathbf{Z}\boldsymbol{\beta}$.

If n experiments are performed and the number of experiments is smaller than the number of coefficients to be determined ($\mathbf{n} < \boldsymbol{k}$) then the regression model is underdetermined and regression analysis cannot be performed. If the number of experiments is equal to the number of coefficients ($\mathbf{n} = \mathbf{k}$), the regression model is determined and the regression equation can be solved exactly rather than approximately as long as all input parameters are linearly independent. In the case where the number of experiments is greater than the number of coefficients ($\mathbf{n} > \mathbf{k}$), the regression model is overdetermined. For the overdetermined systems, the standard approach to determine the polynomial coefficients is the least squares method. The least squares method determines the values ($\mathbf{b_0}, \mathbf{b_1}, \ldots, \mathbf{b_{k-1}}$) for unknown polynomial coefficients such that the final

prediction equation minimizes the sum of the squares of the errors between the actual output from the computer simulations (y) and the predicted output ($\hat{\mathbf{y}}$). Least squares method minimizes the following equation:

$$\mathbf{S(b)} = \sum_{\mathbf{i=1}}^{\mathbf{n}} (\mathbf{y_i} - \hat{\mathbf{y}}_\mathbf{i}(\mathbf{b}))^2 \qquad (\mathbf{3.10})$$

where function S defines the sum of squares, n is the number of experiments, and b is a vector of least squares estimates of coefficients $\boldsymbol{\beta}$. Vector b can be calculated by solving the following matrix equation:

$$\mathbf{b} = (\mathbf{Z'Z})^{-1}(\mathbf{Z'Y}) \qquad (\mathbf{3.11})$$

The final quadratic response surface equation then becomes:

$$\hat{\mathbf{y}} = \mathbf{b_0} + \sum_{\mathbf{i=1}}^{\mathbf{d}} \mathbf{b_i x_i} + \sum_{\mathbf{i=1}}^{\mathbf{d}} \mathbf{b_{ii} x_i^2} + \sum_{\mathbf{i=1}}^{\mathbf{d-1}} \sum_{\mathbf{j>i}}^{\mathbf{d}} \mathbf{b_{ij} x_i x_j} + \boldsymbol{\varepsilon} \qquad (\mathbf{3.12})$$

The process of fitting a response surface to an input-output data set is straightforward as long as a sufficiently large data set is available. However, as the number of input design variables (d) increases, the number of coefficients for the polynomial equations (k) increases. In order to have an overdetermined or determined system, the number of experiments (n) should be equal or larger than the number of coefficients (k). Therefore, in the case of a high number of input design variables, RSM loses its appeal because the number of experiments required to fit the polynomial becomes unmanageable. Moreover, if higher-order polynomials are intended to be used to model non-linear responses, the number of coefficients and the required number of experiments increase compared to the lower-order polynomials with same number of input variables.

For the details of response surface methodology and derivation of least square method equations please refer to (Box and Draper, 1986). MATLAB®, Statistics Toolbox™ is used in this study for RSM least squares regression.

3.3.2 Artificial Neural Networks (ANN)

Artificial neural networks are perhaps the most popular machine learning method. ANNs can efficiently be used as a tool for performing tasks such as function approximation (regression) and classification. ANNs consist of a number of neurons connected to each other with modifiable weighted interconnections and attempts to create a mapping function between the input and output data. ANN is initially trained using a set of experimental data obtained from the computer simulations or laboratory experiments. This set of data is called training data. A supervised learning is used, so training data contains two vectors: an input vector (x) and an output vector (y). ANN learns to relate the input vectors with the output vectors at the end of the training. When it is fed with a previously unseen input vector, ANN is able to interpolate between similar examples in the training data set and to create an output vector. Other than using for function approximation problems with continuous outputs, ANNs can also be used for classification problems. If the output vector is used to represent one of a set of known possible outcomes, the ANN can also act like a classifier.

The basic characteristic of ANN is its architecture. Design of ANN architecture consists of determining the number of layers, the number of neurons in each layer, activation functions of the neurons and the learning algorithm for the network. The most common ANN architecture is a multi-layer feed-forward structure also known as a multi-layer perceptron (MLP) trained by Back-Propagation (BP) algorithm (Kim et al., 1999). There are three different types of layers in a MLP: an input layer representing the input design variables, an output layer representing the response, and a number of hidden layers that performs the mapping of the input data before they enter the output layer. The

term feed-forward means that there are no feedback connections, namely the data only travels from the input layer to the output layer (Paolucci et al., 2000). There are no connections entering the input layer and there are no connections leaving the output layer. Layers consist of neurons such that an interconnection only exists between neurons in adjacent layers. There are no interconnections between the neurons in the same layer. The prediction accuracy of ANNs is highly dependent on the number of hidden layers and the number of neurons in each layer. However, there is no specific rule or procedure to determine the number of hidden layers and neurons which gives the best predictive accuracy. These numbers are usually determined by a trial-error process (Shunmugam and Prasad, 2008; Cho, 2009). If the prediction accuracy of the network is not satisfactory further hidden layers and/or neurons can be added to the network to increase the performance. A simple MLP consisting of three layers: the input layer, the output layer and one hidden layer can be seen in Figure 3.4.

Each neuron (node) in an ANN is a simple processing element which has an input side and an output side (see Figure 3.5). Every neuron can take one or more inputs from one or many other neurons. Each input x_i is multiplied by its associated synaptic weight w_i that determines the intensity of the input. The neuron's function is to sum each of its weighted inputs, add a bias term b to the sum to create an intermediate quantity called the activation, $\boldsymbol{a}$. It then produces an output according to an activation (transfer) function $\boldsymbol{f_t}(.)$. The output obtained from the neuron is then transferred to next neuron or neurons as an input. Most commonly used transfer function is a sigmoid function because of its simple derivative, which is useful for the learning algorithm (see Figure 3.6). The sigmoid function is given by:

$$\boldsymbol{f_t(a) = \frac{1}{1 + e^{-a}}} \qquad (\mathbf{3.13})$$

For a single neuron, the activation $\boldsymbol{a}$ is given by:

$$a = \left(\sum_{i=1}^{d} w_i x_i \right) + b \qquad (\mathbf{3.14})$$

where $\boldsymbol{d}$ is the number of inputs to the neuron and the bias term $\boldsymbol{b}$ is defined separately for each neuron.

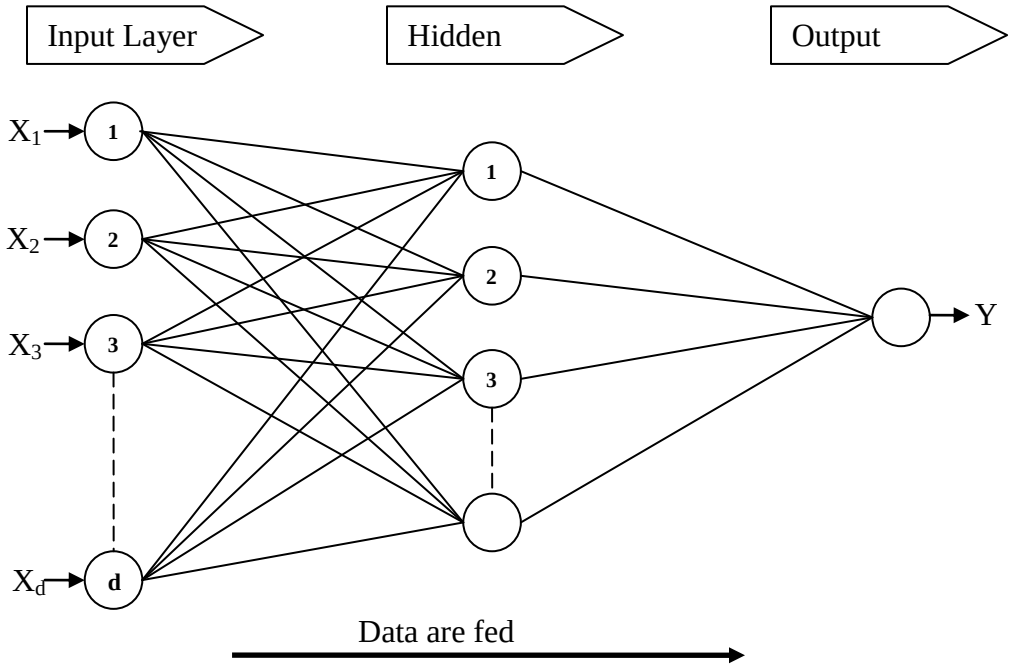


Figure 3.4 A simple Multi-layer Perceptron (MLP) with three layers

There are several learning algorithms developed for training ANNs. The most widely used training algorithm for MLP is the back-propagation (BP) algorithm (Shunmugam and Prasad, 2008). The BP algorithm is a nonlinear extension of the least squares algorithm for MLP (Ferentinou and Sakellariou, 2007). The back-propagation method refers to the way in which the gradient of the squared error function is computed for MLP. BP training consists of two phases: a forward pass where the connection weights

and the biases of the neurons are fixed and backward pass where the weights and biases of the neurons are adjusted in accordance with an error signal.

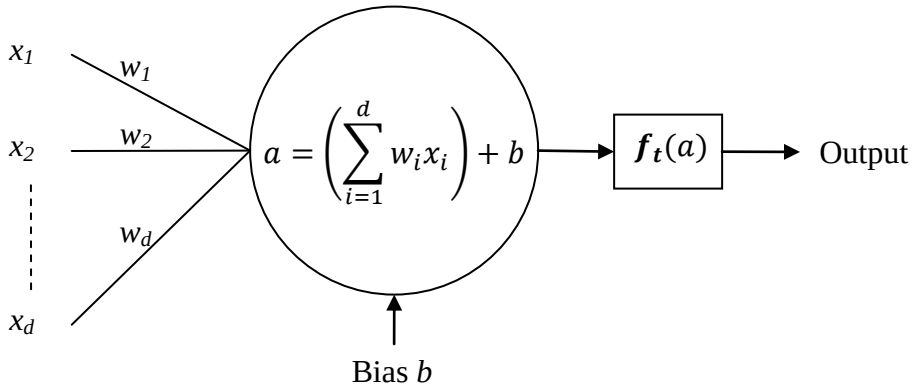


Figure 3.5 A single neuron [Adapted from (Hopgood, 2001)]

Other than the training data, a validation data is used during the learning process. The learning halts when the error of the validation data falls below a threshold value or when a maximum number of iterations (epochs) is reached. Finally, the performance of the network is estimated using an independent test data that has not been used in the learning process. One of the most common mistakes made when using ANN is to use validation data as a performance measure. Validation data cannot be used as a performance measure because even though it is not used directly in the training process the network still learns from the validation data by calibrating its parameters according to validation data's prediction accuracy. The mean square error (MSE) is generally used for calculating the error. A number of algorithms have been developed to increase the speed of convergence for BP training. These algorithms include conjugate gradient and Newton methods which are based on standard optimization techniques. For this study, the Levenberg–Marquardt (LM) algorithm is adopted for its efficiency in training MLP. The details of the BP algorithm can be found in the literature (Haykin, 1999).

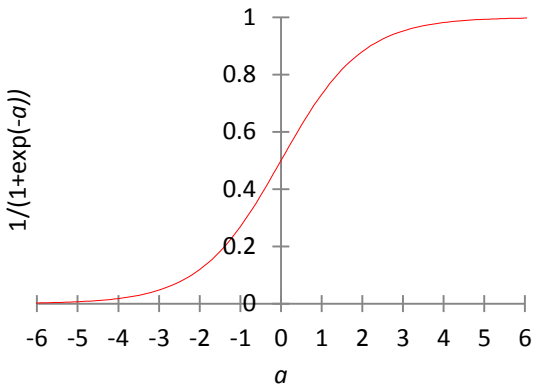


Figure 3.6 Sigmoid function

MATLAB®, Neural Network Toolbox™ is used in this study for all ANN calculations. Specifically, the 'newff' function is used to create feed-forward backpropagation networks for function approximation (regression) problems and the 'newpr' function is used to create pattern recognition networks for classification problems. A customized MATLAB code is written to search for the optimum number of hidden layers and number of neurons that give the best prediction accuracy for both regression and classification problems.

3.3.3 Support Vector Machines (SVM)

The support vector machines (SVM) has recently emerged as a powerful machine learning algorithm based on statistical learning theory (Vapnik, 1995). Rather than using empirical risk minimization, SVMs use structural risk minimization which has been shown to be superior to the more traditional empirical risk minimization used by other classical modeling methods (Dibike et al., 2001). While empirical risk minimization seeks to minimize the classification error only on the training data, structural risk minimization seeks to minimize the bound on the generalization error of the model (Vapnik, 1998). This distinction grants SVM with a better ability to generalize, which is the aim in machine learning. Similar to other machine learning algorithms, SVMs go

through a training phase where the model is presented with a series of input-output data. At the end of the training phase, SVMs are fed with a separate set of data which were never used in the training phase to assess the predictive accuracy of the model. SVMs can be used for classification problems as well as function approximation problems.

3.3.3.1 Support Vector Classification

Support vector machine (SVM) is a maximum margin classifier where it constructs an optimal separating hyperplane that can separate between two different classes of data.

Linearly separable classes can be considered as the simplest case to demonstrate the concept of SVM training. Consider a binary classification problem which has a set of training vectors ($\boldsymbol{D}$) with $\boldsymbol{n}$ number of samples $\{\boldsymbol{x_i}, \boldsymbol{y_i}\}$, $\boldsymbol{i = 1, \dots, n}$, $\boldsymbol{x_i} \in \boldsymbol{R^d}$ represented by two separate classes $\boldsymbol{y_i} \in \{-\mathbf{1}, \mathbf{1}\}$. Assuming $\boldsymbol{D}$ is linearly separable, a linear hyperplane that separates two classes can be written as

$$\mathbf{w}.\mathbf{x} + \boldsymbol{b} = \mathbf{0} \qquad (\mathbf{3.15})$$

where $\mathbf{w}.\mathbf{x}$ denotes the dot product of two vectors, $\mathbf{w}$ is the normal to the hyperplane, $|\boldsymbol{b}|/\|\mathbf{w}\|$ is the perpendicular distance from hyperplane to the origin, and $\|\mathbf{w}\|$ is the Euclidean norm of $\mathbf{w}$. For the simple case of a two dimensional input space ($\boldsymbol{d} = \mathbf{2}$) as shown in Figure 3.7, there are an infinite number of hyperplanes that can linearly separate two classes.

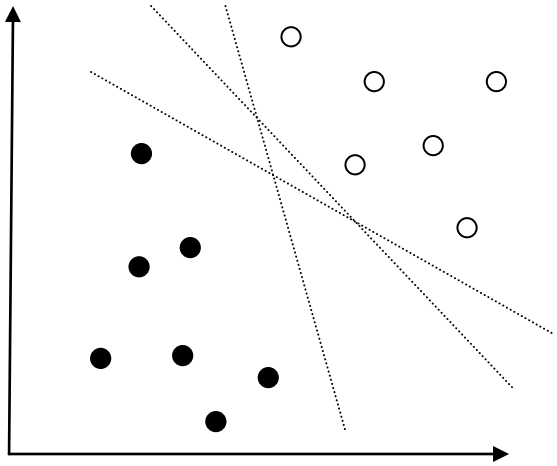


Figure 3.7 Different hyperplanes separating two different classes on a two dimensional input space

As mentioned earlier, SVM is a maximum margin classifier. Margin is defined as the summation of the distances between the hyperplane and the nearest data point of each class. For each hyperplane separating the classes, the width of the margin is different (see Figure 3.8). SVM's objective is to find the optimal separating hyperplane $\boldsymbol{H}$ with the maximum margin in between two parallel supporting planes $\boldsymbol{H_{+1}}$ and $\boldsymbol{H_{-1}}$ (see Figure 3.9). The optimal separating hyperplane minimizes the bound on the actual risk therefore it gives a better generalization performance (Vapnik, 1998).

$$\mathbf{w}.\mathbf{x}_i + \boldsymbol{b} \geq +\mathbf{1} \quad \textbf{for } \boldsymbol{y_i} = +\mathbf{1} \tag{3.16}$$

$$\mathbf{w}.\mathbf{x}_i + \boldsymbol{b} \leq -\mathbf{1} \quad \textbf{for } \boldsymbol{y_i} = -\mathbf{1} \tag{3.17}$$

These two equations can be combined into one set of inequalities:

$$\boldsymbol{y_i}(\mathbf{x}_i.\mathbf{w} + \boldsymbol{b}) - \mathbf{1} \geq \mathbf{0} \quad \forall \boldsymbol{i} \tag{3.18}$$

The points that lie on the hyperplanes $\boldsymbol{H_{+1}}$: $\mathbf{w}.\mathbf{x} + \boldsymbol{b} = \mathbf{+1}$ and $\boldsymbol{H_{-1}}$: $\mathbf{w}.\mathbf{x} + \boldsymbol{b} = \mathbf{-1}$ are the support vectors whose removal would change the solution (Burges, 1998). The distance between these two hyperplanes (margin) is $\mathbf{2}/\|\mathbf{w}\|$.

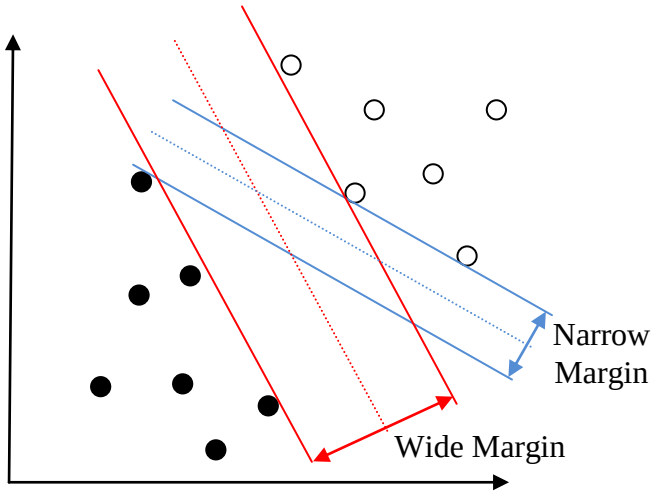


Figure 3.8 Different hyperplanes with different margin width

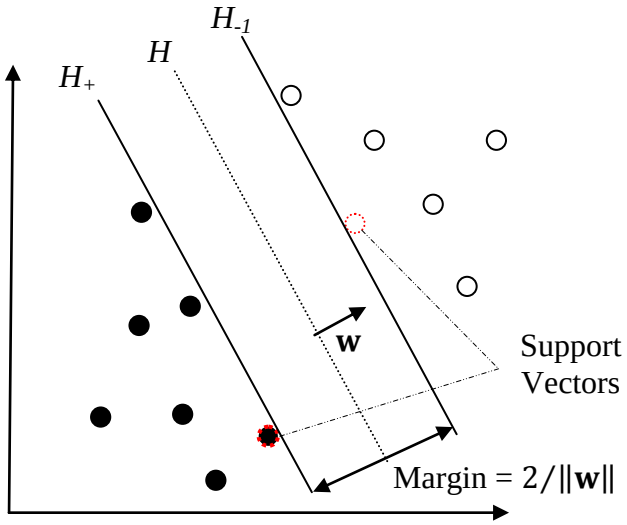


Figure 3.9 Optimal hyperplane H with supporting parallel hyperplanes H_{+1} and H_{-1}

Therefore, the optimal hyperplane is the one that maximizes the margin by minimizing $\|\mathbf{w}\|^2$, subject to constraints in Eq. (3.18). This optimization problem with

constraint can be reformulated into a nonconstrained problem (dual space representation) by using Lagrangian multipliers and then can be solved using standard quadratic programming (QP). By introducing the Lagrange multipliers $\alpha_i,\ i = 1, \dots, n$ the optimization problem can be written as:

$$L_P = \|\mathbf{w}\|^2/2 - \sum_{i=1}^{n} \alpha_i\{[y_i(\mathbf{w}.\mathbf{x}_i) + b] - 1\} \qquad (\mathbf{3.19})$$

The solution is the saddle point of this Lagrange function which can be determined by differentiating L_P with respect to **w** and *b* and setting the results equal to zero:

$$\frac{\partial L}{\partial \mathbf{w}} = 0 \ \rightarrow\ \mathbf{w} = \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i \qquad (\mathbf{3.20})$$

$$\frac{\partial L}{\partial b} = 0 \ \rightarrow\ \sum_{i=1}^{n} \alpha_i\, y_i = 0 \qquad (\mathbf{3.21})$$

Since these are equality constraints in the dual formation, they can be substituted into (3.19) to give the dual form equation which has to be maximized with the constraint $\alpha_i \geq 0$:

$$L_D = \sum_i \alpha_i - \frac{1}{2}\sum_{i,j} \alpha_i \alpha_j y_i y_j \mathbf{x}_i.\mathbf{x}_j \qquad (\mathbf{3.22})$$

L_P (P for primal) and L_D (D for dual) are the result of same objective function but with different constraints. There is a Lagrange multiplier α_i for every training point. After solving Eq. (3.22) using quadratic programming (QP) for Lagrange multipliers, those training points with $\alpha_i > 0$ are called "support vectors" and they lie on one of the supporting hyperplanes H_{+1} or H_{-1}. For SVM training, support vectors are the critical elements of the training set. If training points other than support vectors are to be

removed from the training set, the solution for the separating hyperplane would be the same (Burges, 1998).

The above solution can only be used if the training data is linearly separable. In the case of linearly inseparable data, the concept of "soft margin" is introduced to solve the problem (Cortes and Vapnik, 1995). Soft margin concept relaxes the restriction (Eq. (3.4) and Eq. (3.5)) that every training vector of a certain class should be on the same side of the optimal separating hyperplane (see Figure 3.10). This is achieved by introducing positive slack variables $\boldsymbol{\xi_i \geq 0,\ i = 1, \ldots, n}$, then Eq.(3.4) and Eq.(3.5) become:

$$\mathbf{w}.\mathbf{x_i} + \boldsymbol{b} \geq +\mathbf{1} - \boldsymbol{\xi_i} \quad \textbf{for } \boldsymbol{y_i} = +\mathbf{1} \tag{3.23}$$

$$\mathbf{w}.\mathbf{x_i} + \boldsymbol{b} \leq -\mathbf{1} + \boldsymbol{\xi_i} \quad \textbf{for } \boldsymbol{y_i} = -\mathbf{1} \tag{3.24}$$

These two equations can be combined into one set of inequalities:

$$\boldsymbol{y_i}(\mathbf{x_i}.\mathbf{w} + \boldsymbol{b}) \geq \mathbf{1} - \boldsymbol{\xi_i} \quad \forall \boldsymbol{i} \tag{3.25}$$

The objective now becomes both maximizing the margin of the hyperplane and at the same time minimizing misclassification error introduced by slack variables. The result of this objective is an optimization problem where the goal is minimizing the following equation with constraint in Eq. (3.25):

$$\frac{\mathbf{1}}{\mathbf{2}}\|\mathbf{w}\|^{\mathbf{2}} + \boldsymbol{C}\sum_{\boldsymbol{i=1}}^{\boldsymbol{n}} \boldsymbol{\xi_i} \tag{3.26}$$

where $\boldsymbol{C}$ $(\infty > \boldsymbol{C} > \mathbf{0})$ is a parameter chosen before training that defines the cost of the constraint violation. This parameter controls the trade-off between the maximization of margin and the minimization of number of misclassifications. A small *C* assigns a small penalty to misclassification errors, this leads to maximization of the margin even with the large number of errors and a less complex model. A large *C* gives a large penalty to the

misclassification errors, therefore SVM tries to minimize the errors other than generalization and this leads to a more complex model. Parameter *C* is one of the two parameters that needs to be calibrated to increase the predictive accuracy of SVM. It is usually determined by trial and error with the use of a training and test data set.

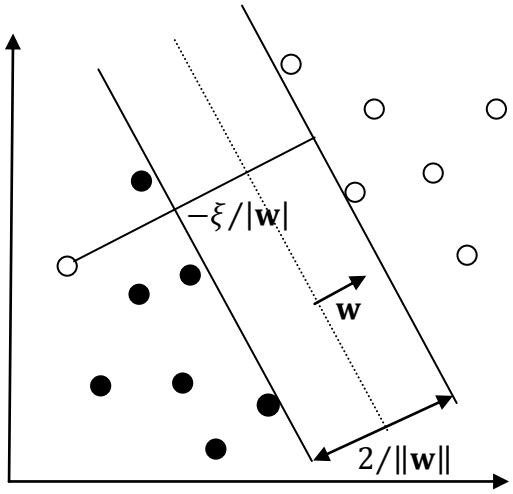


Figure 3.10 Linearly inseparable data and soft margin

The solution to this optimization problem is similar to the linearly separable case where Lagrangian multipliers and quadratic programming were used. The primal Lagrangian is written as follows:

$$L_P = \|\mathbf{w}\|^2/2 + C\sum_{i=1}^{n}\boldsymbol{\xi_i} - \sum_{i=1}^{n}\alpha_i\{[y_i(\mathbf{w}.\mathbf{x}_i) + b] - 1 + \boldsymbol{\xi_i}\} - \sum_{i=1}^{n}\mu_i\boldsymbol{\xi_i} \qquad (\mathbf{3.27})$$

where μ_i are the Lagrange multipliers introduced to enforce the positivity of $\boldsymbol{\xi_i}$. And the dual problem is written as the same equation with linearly separable case except with different constraints in Eq. (3.29):

$$L_D = \sum_i \alpha_i - \frac{1}{2}\sum_{i,j}\alpha_i\alpha_j y_i y_j \mathbf{x}_i.\mathbf{x}_j \qquad (\mathbf{3.28})$$

$$0 \leq \alpha_i \leq C \quad \text{and} \sum_{i=1}^{n} \alpha_i \, y_i = 0 \qquad (\mathbf{3.29})$$

Solving Eq.(3.28) with quadratic programming (QP) determines the Lagrange multipliers. Unlike neural networks, no local minima can occur in this optimization problem, therefore the SVM gives a unique optimal separation plane (Goh and Goh, 2007).

In the case where the decision function is not linear, the input space can be mapped into a higher dimensional feature space using a nonlinear mapping function **Φ** (Boser et al., 1992). After mapping the training data-set into a higher dimensional feature space, the training data may be moved away from each other and a larger margin may be found for the optimal separating hyperplane (see Figure 3.11).

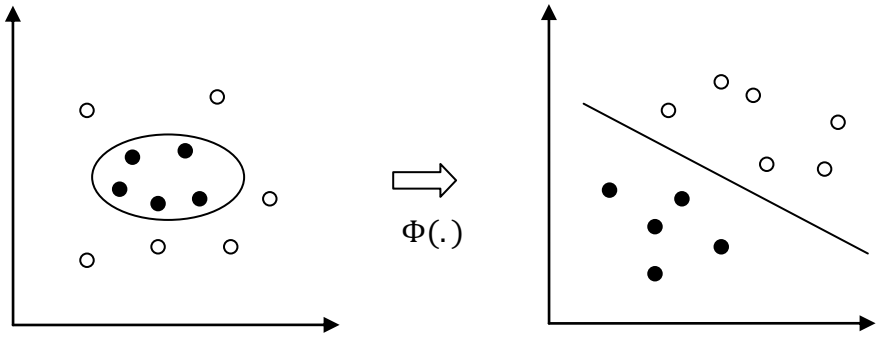


Figure 3.11 Non-linear mapping from input space to feature space [Adapted from Goh and Goh (2007)]

After the introduction of the mapping function **Φ** Eq. (3.28) becomes:

$$L_D = \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j . (\Phi(\mathbf{x}_i) . \Phi(\mathbf{x}_j)) \qquad (\mathbf{3.30})$$

with the same constraints as in Eq. (3.29). Next, a "kernel function" *K* is introduced such that:

$$\boldsymbol{K}(\mathbf{x}_i, \mathbf{x}_j) = \sum_{i,j} \boldsymbol{\Phi}(\mathbf{x}_i).\boldsymbol{\Phi}(\mathbf{x}_j) \qquad (\mathbf{3.31})$$

Introducing the kernel function K enables the training to be performed without explicitly knowing $\boldsymbol{\Phi}$ therefore reducing the computational demand (Boser et al., 1992). There are different kernel functions that can be used with SVM. Most commonly used kernel functions are simple polynomial and radial basis kernel functions (RBF). Simple polynomial kernel function can be written as:

$$\boldsymbol{K}(\mathbf{x}_i, \mathbf{x}_j) = ((\mathbf{x}_i.\mathbf{x}_j) + \mathbf{1})^{\boldsymbol{d}} \qquad (\mathbf{3.32})$$

where $\boldsymbol{d}$ is the degree of the polynomial and needs to be defined by the user before the training process. Radial basis kernel function (RBF) can be written as:

$$\boldsymbol{K}(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{exp}\left(-\boldsymbol{\gamma}|\mathbf{x}_i - \mathbf{x}_j|^2\right), \qquad \boldsymbol{\gamma} > \mathbf{0} \qquad (\mathbf{3.33})$$

where $\boldsymbol{\gamma}$ is the kernel parameter and it is also user defined. Radial basis kernel function is used in this study for constructing SVMs.

With the choice RBF, there are only two parameters needed to be assigned before the training of SVM: a penalty parameter *C* and the RBF parameter $\boldsymbol{\gamma}$. The objective of the SVM training is to find the combination of these two parameters that gives best prediction accuracy on a validation data. The optimum values for C and $\boldsymbol{\gamma}$ are not known prior to training therefore a parameter search must be performed to find these values. Sophisticated models such as evolutionary algorithms can be used to search for the optimum parameters (Feng et al., 2004). The grid search algorithm, a relatively simple method, is adopted in this study to find the optimum parameters. In the gird search method, the possible intervals of C and $\boldsymbol{\gamma}$ are provided by the user and training is performed on each pair of the values. Performing a complete grid search is time consuming. Therefore, a grid search is first performed on a coarser grid and after

determining the region which gives better prediction accuracy, a finer grid search is performed on that region. After determining the values of C and $\boldsymbol{\gamma}$ that gives the best prediction accuracy on validation set, the final performance of the model is estimated using a separate test set that has never been used in the learning process. The details of the grid search method will be demonstrated in Section 3.5.

For all the cases presented above, SVM was used for separating only two different output classes. The original algorithm of SVM can only handle two-class outputs. However, it can easily be modified to obtain classifiers with more than two-class outputs. Consider a classification problem where the number of output classes *k*, is larger than two. In order to solve this problem, a number of *k* SVM classifiers can be created to separate each class from the rest. Then to predict the output class of a new input, input is fed into each classifier and by applying a voting scheme the class is determined (Dibike et al., 2001).

3.3.3.2 Support Vector Regression

Initial research on SVM focused on classification problems and SV classifiers has become competitive with the best available machine learning algorithms. However, excellent performances were soon obtained also in function approximation (regression) problems (Smola and Scholkopf, 2004). There are two different approaches to SVM regression: ε-SV regression (Vapnik, 1995) and v-SV regression (Scholkopf et al., 2000). In this study, ε-SV regression is adopted.

Consider a regression problem which has a set of training vectors ($\boldsymbol{D}$) with $\boldsymbol{n}$ number of samples $\{\boldsymbol{x_i}, \boldsymbol{y_i}\}$, $\boldsymbol{i = 1, \dots, n}$, $\boldsymbol{x_i \in R^d}$, $\boldsymbol{y_i \in R}$. The objective of ε-SV regression is to find a function $\boldsymbol{f}(\mathbf{x})$ that has at most ε deviation from the actual output values and at the same time as flat as possible (Smola and Scholkopf, 2004). Similar to the soft margin concept in SVM classification, the restriction on the deviation imposed by ε can be relaxed by introducing slack variables $\boldsymbol{\xi_i, \xi_i^* \geq 0}$ (see Figure 3.12). The result

of this objective function is an optimization problem where the aim is to minimize Eq. (3.34) subject to constraints in Eq. (3.35):

$$\frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n}(\xi_i + \xi_i^*) \tag{3.34}$$

$$y_i - \mathbf{w}.\mathbf{x}_i - b \leq \varepsilon + \xi_i \tag{3.35a}$$

$$\mathbf{w}.\mathbf{x}_i + b - y_i \leq \varepsilon + \xi_i^* \tag{3.35b}$$

where C is the penalty parameter for the slack variables. For the nonlinear case, kernel functions are used to map the input space to a higher dimensional feature space. Similar to SVM classification, this optimization problem can be solved by introducing Lagrange multipliers and quadratic programming (QP). For a detailed tutorial on SVM regression please refer to Smola and Scholkopf (2004).

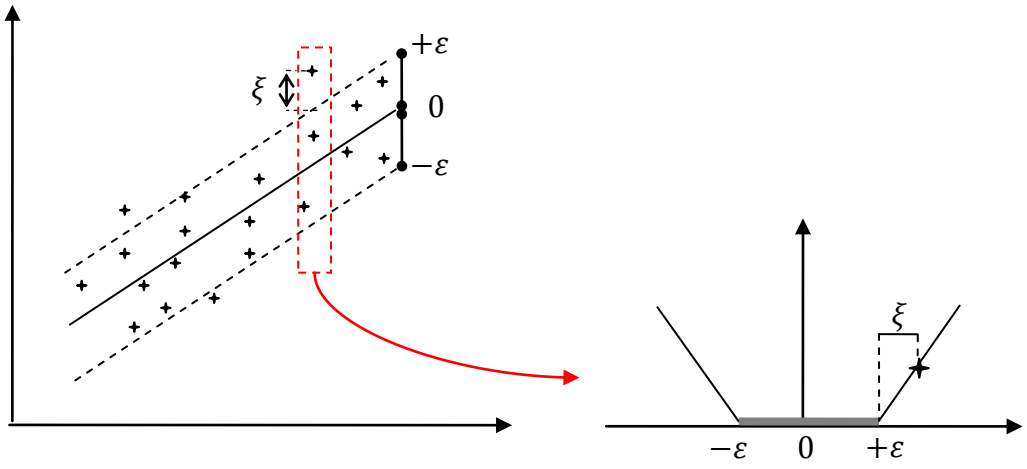


Figure 3.12 Specified deviation $\boldsymbol{\varepsilon}$ and slack variable $\boldsymbol{\xi}$ in ε-SV regression [Adapted from Smola and Scholkopf (2004)]

Radial basis function kernel in Eq. (3.33) is used in this study for the nonlinear mapping function for ε-SV regression. Unlike SVM classification, there are three parameters needed to be assigned before the training: a penalty parameter C, a deviation parameter ε, and the RBF parameter $\boldsymbol{\gamma}$. The aim of the SVM training is to find the combination of these three parameters that gives best prediction accuracy. A parameter search similar to SVM classification must be performed to find the optimum values of these three parameters.

There are several advantages of SVM compared to ANN for both classification and regression problems. The solution to optimization problem in SVM is unique, global and deterministic. For the same parameter combinations (C, ε, $\boldsymbol{\gamma}$) and same training set the same solution is always found. On the other hand, even if the network structure is the same, ANN solutions arrive at a local minimum and a different solution each time the network is initiated with initial weights and biases. SVM is less vulnerable to overfitting problems than ANN. Very few model parameters are used to calibrate the SVM models compared to ANN.

Several computer codes are available for SVM modeling. In this study, for both SVM classification and SVM regression problems, the computer code LIBSVM is used (Chang and Lin, 2001). LIBSVM is available for several different computer languages. LIBSVM Version 3.0 for MATLAB® is used for this study.

3.4. Validating Metamodels

Metamodel validation is one of the most important steps in the metamodel building process. In validation of the prediction accuracy of the metamodel, it is widely known that performance measures of the model are biased if they are calculated from the same set of data that is used to train the model. If the model is trained just by maximizing its performance on the training data set, there is a big possibility of overfitting. An

overfitted model only memorizes the trends in training data set rather than learning to generalize and performs poorly on previously unseen test data.

The simplest way to overcome the overfitting problem and measuring the performance of the model more precisely is to divide the data set into two parts: learning set and test set. Most metamodeling techniques such as ANNs and SVMs require user to calibrate certain parameters (such as the penalty parameter *C*, and the RBF parameter $\boldsymbol{\gamma}$ for SVMs or the number of layers and neurons for ANNs). Therefore, for the calibration of these control parameters, the learning set may further be divided into a training set and a validation set (Molinaro et al., 2005). The training set is used to train the metamodel. After fitting the metamodel on the training set, the metamodel is used to predict the values of the validation data for which the true outputs are known but not used in training the model. The predicted and true values of the validation data give a less biased estimate of the error rate of the metamodel (Michie et al., 1994). The validation set is used to calibrate the parameters that give the best prediction accuracy. However, when a model with optimal parameters is chosen, its prediction accuracy with the validation dataset is still a biased estimate because the model parameters are chosen based on the validation set. Namely, the validation set is involved in the learning processes. Therefore, there is a need for another completely independent portion of data which has never been used in both training and validation. This portion of data is called the test set. The test set is only used to assess the performance of a fully-trained metamodel. After assessing the performance of the final model with the test set, the model parameters must not be changed further to avoid bias. Different performance measures can be used to calculate the performance of the metamodel. In this study, for classification problems the model accuracy is calculated with the following formula:

$$\left(\frac{\boldsymbol{Number\ of\ data\ points\ accurately\ predicted}}{\boldsymbol{Total\ number\ of\ data\ points}}\right) \boldsymbol{X}\ \mathbf{100} \qquad (\mathbf{3.36})$$

For regression problems, the squared correlation coefficient (R^2), mean squared error (MSE) and mean absolute percentage error (MAPE) are used as performance measures. These parameters are calculated as follows:

$$\mathbf{R^2} = \frac{(\boldsymbol{n}\sum_{i=1}^{n}\hat{\boldsymbol{y}}_i\boldsymbol{y}_i - \sum_{i=1}^{n}\hat{\boldsymbol{y}}_i\sum_{i=1}^{n}\boldsymbol{y}_i)^2}{\left(\boldsymbol{n}\sum_{i=1}^{n}\hat{\boldsymbol{y}}_i^{\,2} - (\sum_{i=1}^{n}\hat{\boldsymbol{y}}_i)^2\right)\left(\boldsymbol{n}\sum_{i=1}^{n}\boldsymbol{y}_i^{\,2} - (\sum_{i=1}^{n}\boldsymbol{y}_i)^2\right)} \qquad (\mathbf{3.37})$$

$$\mathbf{MSE} = \frac{\mathbf{1}}{\boldsymbol{n}}\sum_{i=1}^{n}(\hat{\boldsymbol{y}}_i - \boldsymbol{y}_i)^2 \qquad (\mathbf{3.38})$$

$$\mathbf{MAPE} = \frac{\mathbf{1}}{\boldsymbol{n}}\sum_{i=1}^{n}\left|\frac{\hat{\boldsymbol{y}}_i - \boldsymbol{y}_i}{\boldsymbol{y}_i}\right| \qquad (\mathbf{3.39})$$

where $\hat{\boldsymbol{y}}_i$ is the predicted value, $\boldsymbol{y}_i$ is the actual value and $\boldsymbol{n}$ is the number of data points.

For moderate sized samples, other than just using one learning set and one test set, the *k*-fold cross-validation method can be used. In the *k*-fold cross-validation method the whole set is divided into *k* subsets of equal size. Each subset is then used as a test set with the model trained with the remaining (*k*-1) subsets. The cross-validation process is repeated *k* times with each of the subsets which is used exactly once as the test data. The predictive accuracy of the model is then calculated by averaging the performance measures from *k* subsets. The choice of $k = 10$ is the most popular for *k*-fold cross-validation (Kim, 2009). A variation of *k*-fold cross-validation is stratified *k*-fold cross-validation. In stratified *k*-fold cross-validation, the folds are constructed so that the mean output of each fold is the same as the initial dataset. For the case of classification problems, this means that each fold contains roughly the same proportions of the types of classes. Empirical studies show that stratified cross-validation produces more stable estimations of the predictive model performances because it reduces the variance (Diamantidis et al., 2000)

Even though it is known to be nearly unbiased, the *k*-fold cross-validation has a variation due to the randomness of partitioning the sample into *k*-folds (Kim, 2009). This variance can be reduced by repeating the process of partitioning the data into stratified folds and estimating the performance. The data set is reshuffled and re-stratified before each repetition and the average performance of all repetitions is taken as the final estimate of the performance. This method is called repeated cross-validation (Diamantidis et al., 2000; Kim, 2009). The number of repetitions reported in the literature is in the range of 5 to 10 (Braga-Neto and Dougherty, 2004; Kim, 2009) .

Cross-validation can also be used to calibrate the model parameters. For the case of SVM classification, instead of dividing the learning data set into just one training and validation data set, cross-validation can be used to determine the optimum penalty parameter *C*, and the RBF parameter $\boldsymbol{\gamma}$. In the grid search, for each parameter combination the SVM classifier is trained and tested on all folds, then the average prediction accuracy is assigned to that parameter combination. Finally the optimum parameters that give the highest cross-validation prediction accuracy are used to train the whole learning set. The optimum parameters can be affected by the size of data set but in practice the ones that are obtained from cross-validation are already suitable for the whole data set (Chang and Lin, 2001).

3.5. Feasibility Study and Comparison of Different Metamodeling Techniques

In this section of the thesis, a feasibility study is presented to explore the effectiveness of metamodeling process for approximating deterministic computer experiments. The static factor of safety calculation of a homogenous dry slope is chosen as a relatively simple, yet realistic example to demonstrate the steps involved in metamodeling and familiarize the reader with the process of creating metamodels. Predicting the factor of safety of slopes is approached as both a functional approximation (regression) problem where the aim is to predict the static factor of safety and a

classification problem where the slope is defined as either stable or unstable. Different metamodeling techniques are investigated. In particular, for the functional approximation problem, the relatively new machine learning methods of Support Vector Machines (SVM) and Artificial Neural Networks (ANN) are compared to the more common Response Surface Methodology (RSM). For the classification problem, ANN and SVM are compared to commonly used Naïve Bayes classifiers. This example sets the stage for using these soft computing methods in the case of dynamic response analyses of slopes.

3.5.1 Design of Experiments for the Static Factor of Safety Problem

Five parameters (design variables) are chosen for calculating the static factor of safety of homogenous dry slopes. The parameters selected, as shown in Figure 3.13, include the height of the slope ($\boldsymbol{H}$), the inclination of slope ($\boldsymbol{\beta}$), the density of soil ($\boldsymbol{d}$), the effective cohesion of soil ($\boldsymbol{c'}$) and the effective friction angle of soil ($\boldsymbol{\phi'}$). The parameter ranges selected for design variables are shown in Table 3.1.

A sequential Latin hypercube sampling is used to generate the set of computer experiments. First, a run size of 50 experiments is created using a *maximin* Latin hypercube design. "Latin Hypercube Samples" package is used with "R" statistical software for all the experimental designs mentioned in this study.

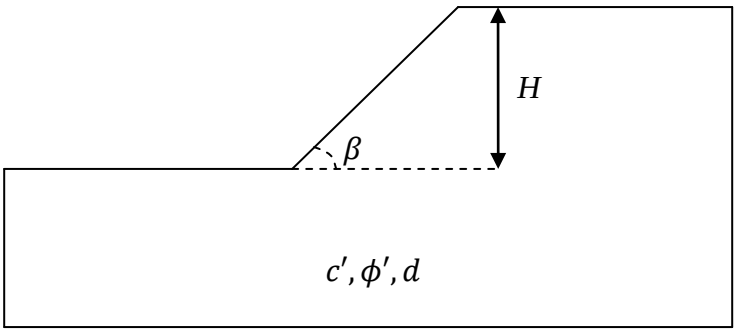


Figure 3.13 Simple homogenous dry slope with design variables

This initial set is not included in the training of the models and it is only used for testing the predictive accuracy of the metamodels. A scatterplot matrix of two dimensional projections of 5 design variables for 50 input points scaled to [0, 1] can be seen in Figure 3.14. As it can be seen from the figure the maximin Latin hypercube design has good space filling properties.

After creating the initial test set, a training set with a size of 30 experiments are added to the original input space without disturbing the Latin characteristic of the design using augmented Latin hypercube tool. Following the augmentation of 30 training points, 10 more points are added for the validation set. Therefore, a total of 40 learning set points are added to the original input space. A scatterplot matrix of 40 augmented learning set points can be seen in Figure 3.15. In the figure, red plus signs correspond to the augmented 40 learning set points and blue dots that represent the original input space with 50 test points. The augmented learning set points are uniformly distributed and have good space filling properties. Moreover, augmented learning set points do not intersect with the original test points (no replication) and fill the spaces in between test points.

Subsequently, more learning set points are augmented to the original space in the increments of 30 training and 10 validation points until there is no further improvement in the model performances. Therefore, the performance of the different metamodels can be evaluated with respect to the size of the learning data set.

Table 3.1 Assigned ranges for design variable values

	Design Variables				
	H	β	d	c'	ϕ'
Lower bound	10 m	10^o	1600 kg/m^3	5 kPa	20^o
Upper bound	350 m	60^o	2500 kg/m^3	50 kPa	50^o

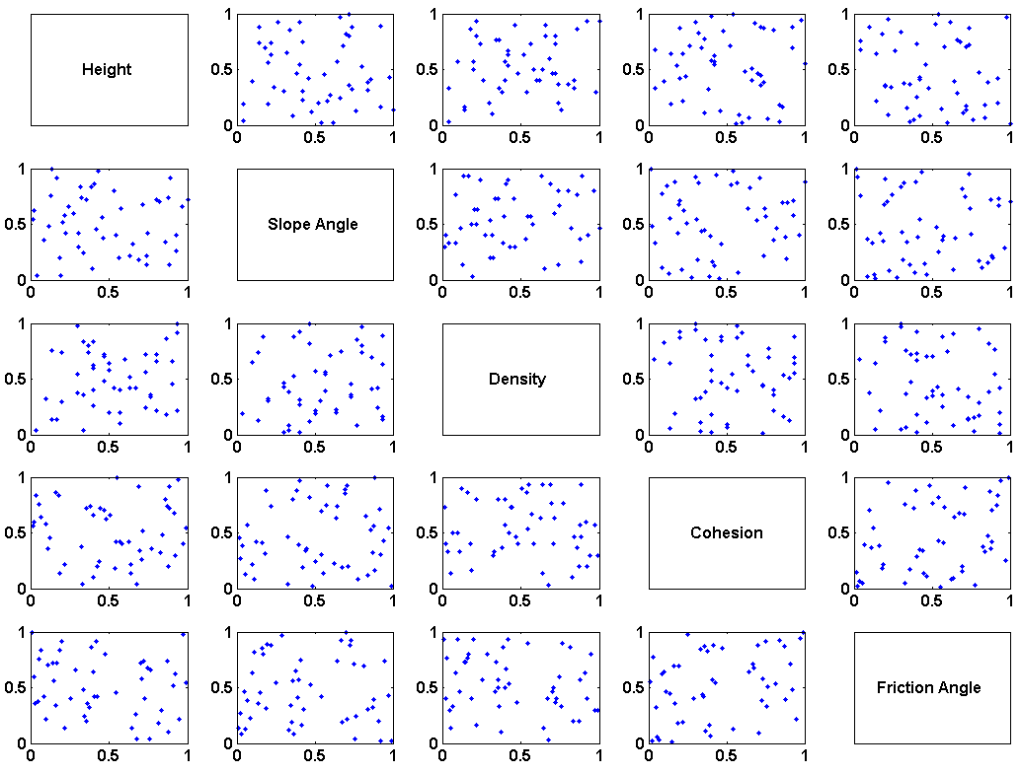


Figure 3.14 A scatterplot matrix of 50 test points

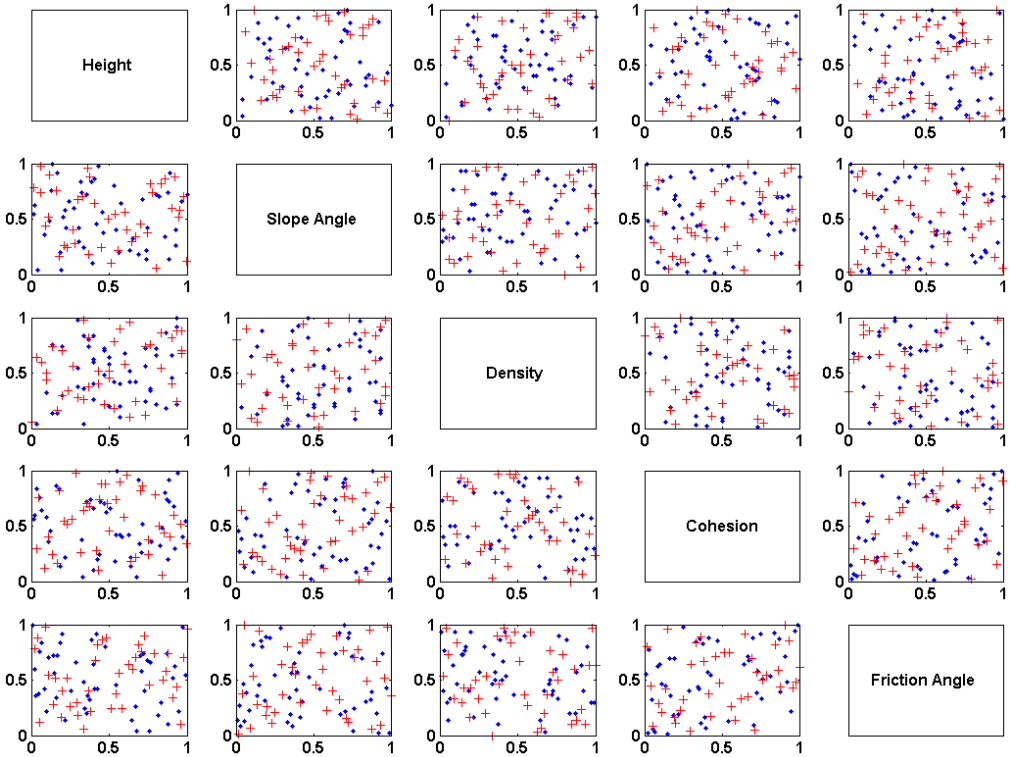


Figure 3.15 A scatterplot matrix of 50 test points plus 40 learning points

3.5.2 Static Factor of Safety Calculations by the Strength Reduction Method

After all the learning and testing experiments are designed by the augmented Latin hypercube model, computer simulations are performed to calculate the static factor of safety for each experiment. The commercial finite difference method (FDM) package FLAC (ITASCA, 2008) is used to perform the computer simulations. FLAC provides an alternative to commonly used limit equilibrium methods of slices. It uses the strength reduction method (SRM) by progressively reducing or increasing the shear strength of the material to bring the slope to a state of limiting equilibrium (Zienkiewicz et al., 1975). Unlike limit equilibrium methods, SRM makes no assumptions on the interslice forces and there is no need to specify a range of trial failure surfaces before the analysis. Failure modes and surfaces develop naturally (see Figure 3.16).

All of the designed experiments are solved using an elastic-perfectly plastic Mohr-Coulomb model with no hardening or softening. A non-associated flow rule is used. The computer run times for the factor of safety calculations using strength reduction method by FLAC depends on the number of elements in the model. The number of elements depends on the geometry of the slope. For example, for a 100 meters high slope with 6400 elements, factor of safety calculation takes about 20 minutes on a 2.8 GHz Intel® Core™ i7 computer.

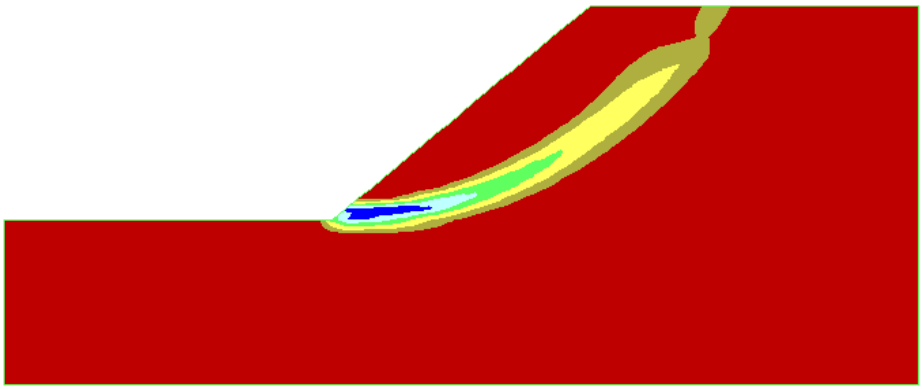


Figure 3.16 A sample FLAC analysis using SRM method

3.5.3 Static Factor of Safety as a Regression Problem

In this section, the static factor of safety of homogenous dry slopes is considered as a function approximation (regression) problem. Response surface methodology (RSM), artificial neural networks (ANN) and support vector machines (SVM) are used as metamodeling techniques to predict the static factor of safety. All of the three models are trained and tested with the same sets of learning and testing data created with the augmented Latin hypercube design, therefore the performances of the different techniques can be compared. As part of the sequential metamodeling approach, the performances of the models are examined while increasing the number of learning points.

3.5.3.1 Response Surface Methodology Approach

A quadratic response function similar to Eq. (3.4) which takes into account individual parameter effects, second order effects, and two parameter interactions is used for fitting the training data. The performance of the model is evaluated using the test data. MATLAB® 'regstats' function is employed to fit the response surface using the least squares regression method. Since there are 5 design variables $(\boldsymbol{\beta}, \boldsymbol{H}, \boldsymbol{\phi}', \boldsymbol{c}', \boldsymbol{d})$ in the input design space, the number of coefficients to be determined in the quadratic response function is 21 including a constant term. Therefore in order to have a determined $(\mathbf{n} = \mathbf{k})$ or an overdetermined $(\mathbf{n} > \mathbf{k})$ system, the number of training points should be equal or greater than 21. Unlike ANN and SVM, there are no parameters to calibrate for RSM; therefore there is no need to use a validation dataset. All of the learning sets are used to fit the response surfaces. The first batch of 40 learning points is used to fit the RSM model and 50 test points are used to evaluate the performance of the model. The resulting quadratic response surface equation for the factor of safety (F.S.) is:

$$F.S.(\beta, H, \phi', c', d) = -3.17968 - 0.08003\beta - 0.00829H + 0.14593\phi' + 8.18\text{x}10^{-5}c' + 0.002938d - 5.88\text{x}10^{-5}\beta H - 0.00162\beta\phi' + 7.15\text{x}10^{-7}\beta c' - 9.43\text{x}10^{-6}\beta d + 4\text{x}10^{-5}H\phi' - 7.81\text{x}10^{-8}Hc' + 3.41\text{x}10^{-6}Hd - 4.45\text{x}10^{-7}\phi' c' - 2.42\text{x}10^{-5}\phi' d - 4.30\text{x}10^{-8}c' d + 0.001317\beta^2 + 7.13\text{x}10^{-6}H^2 + 9.84\text{x}10^{-6}\phi'^2 + 3.29\text{x}10^{-10}c'^2 - 2.70\text{x}10^{-7}d^2 \qquad (3.40)$$

Figure 3.17 shows the performance of the RSM model trained with 40 learning set points. The actual values of factor of safety obtained from the computer simulations and the corresponding predicted values from response surface model are shown for both the training and the testing data sets. As can be seen in the figure, even though the model fits the training data very well (MSE=0.0032), it performed poorly with the test data (MSE=0.058).

The same procedure is followed while sequentially increasing the number of learning points in increments of 40 up to a total of 200 experiments. The performance of the RSM model is evaluated with respect to the number of training points at each increment. The plots of actual versus predicted values for every increment can be seen in Appendix A (Figure A.1-A.3). Table 3.2 summarizes the performance of RSM model for increasing numbers of learning points.

As can be seen in Table 3.2, initially the performance of RSM increases significantly with increasing the number of training points from 40 to 80, however the performance doesn't change notably after adding more training points. The model is saturated after this point and it learnt as much as it could from the data. Adding more points only causes small fluctuations in the performance of the model. Figure 3.18 shows the best performed RSM model which is trained with 80 learning set points.

Table 3.2 Performance summary of RSM with different number of training points

Number of Training Points	Training			Testing		
	R^2	MSE	MAPE	R^2	MSE	MAPE
40	1	0.0032	4.946	0.96	0.058	35.94
80	0.99	0.016	8.64	0.99	0.019	8.998
120	0.99	0.017	10.16	0.98	0.023	10.29
160	0.99	0.02	11.26	0.98	0.025	9.856
200	0.99	0.02	11.59	0.98	0.022	9.733

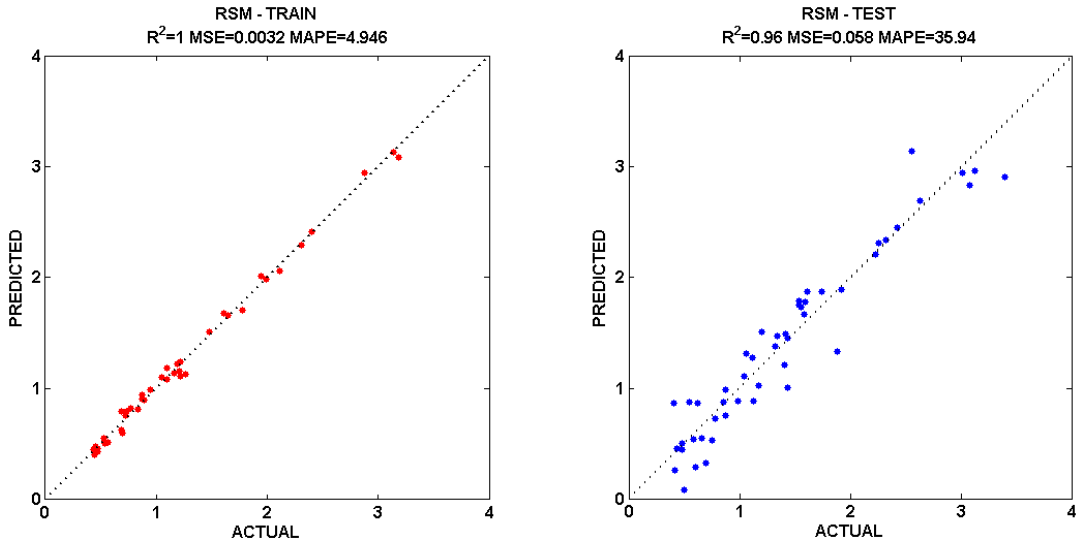


Figure 3.17 Performance of RSM model for the first batch of 40 learning points

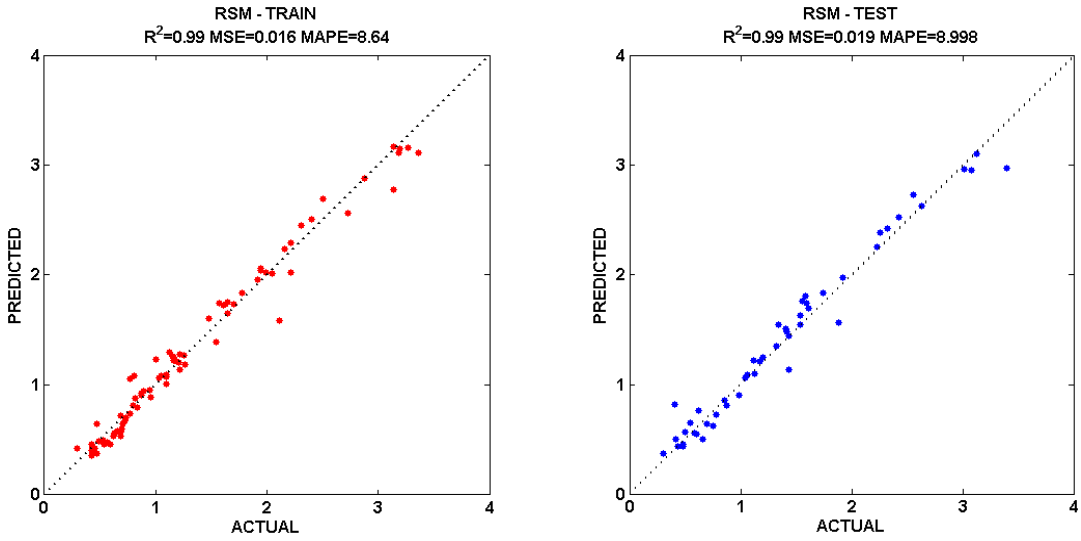


Figure 3.18 Performance of RSM model for 80 learning points

3.5.3.2 Artificial Neural Networks Approach

A multi-layer feed-forward neural network structure trained by the Back-Propagation (BP) is used for the static factor of safety regression problem. Levenberg–Marquardt (LM) algorithm is adopted for the training. The network is trained with the learning data set and the performance of the network is evaluated by the use of the test data set. The mean square error (MSE) is used as a performance measure. MATLAB®, Neural Network Toolbox™ 'newff' function is employed to create three layer feed-forward networks (see Figure 3.19). The first layer consists of the 5 design variables ($\boldsymbol{\beta}, \boldsymbol{H}, \boldsymbol{\phi}', \boldsymbol{c}', \boldsymbol{d}$) in the input design space. The second layer is the hidden layer. The number of the neurons (N) in the hidden layer is decided by a trial and error process where the objective is to find the optimal number of neurons which gives the best prediction performance on the validation data set. The third layer is the output layer which is a vector of the factor of safety values (F.S.) corresponding to each experiment. A sigmoid transfer function is used for the hidden layer and a linear transfer function is used for the output layer. Before training, both inputs and outputs are scaled so that they fall in the range [-1, 1]. The reason for scaling is to avoid inputs in greater numeric ranges dominating the inputs in smaller numeric ranges. An additional advantage for scaling is to avoid numerical difficulties during training (Chang and Lin, 2001). Unlike support vector machines, neural networks might produce different solutions each time the network is initiated (Demuth et al., 2009). Therefore, during the training process the network is initiated numerous times, and finally the network which gives the best performance is chosen.

The first batch of 40 learning points is used to train the ANN model. Out of 40 learning points 30 points are used to train the network and 10 points are used for validation. 50 test points are used to evaluate the performance of the model. 10 neurons are used in the hidden layer. The performance of the ANN model for the first 40 learning points can be seen in Figure 3.20. ANN model fits the training and validation data well

(MSE = 0.0088 and MSE = 0.013). It performs slightly better (MSE = 0.055) than the RSM model (MSE = 0.058) on test data.

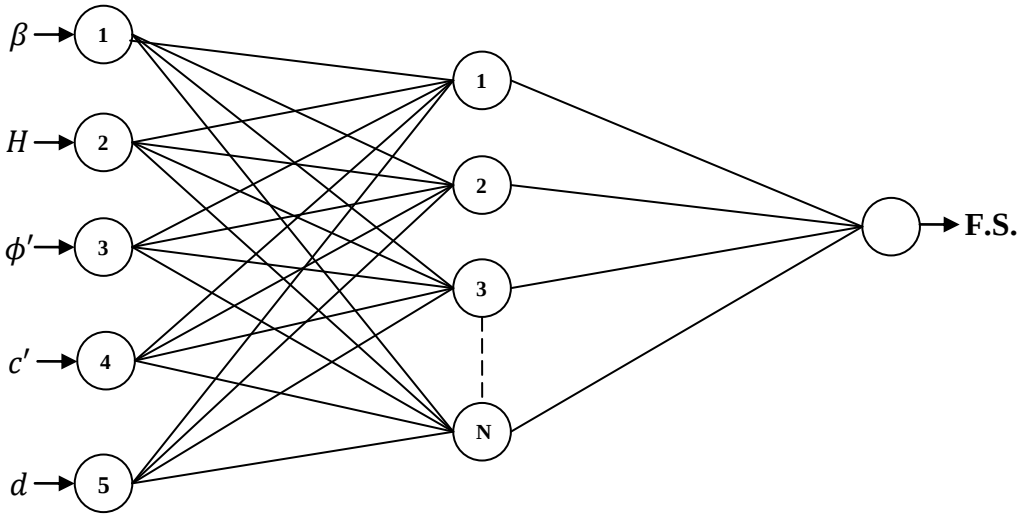


Figure 3.19 ANN structure for the static factor of safety regression problem

The same procedure is followed to train and test the ANN models for increasing number of learning points. The performance plots showing the actual versus predicted values for each increment of increasing numbers of learning points can be seen in Appendix A (Figure A.4-A.6). A summary of the performances of ANN models for increasing numbers of learning points can be seen in Table 3.3.

As can be seen in Table 3.3, the performance of the ANN model improves with increasing number of learning points. The maximum performance is obtained at around 160 learning points and afterwards adding more learning points doesn't improve the performance. The performance of the ANN model for 160 learning points can be seen in Figure 3.21.

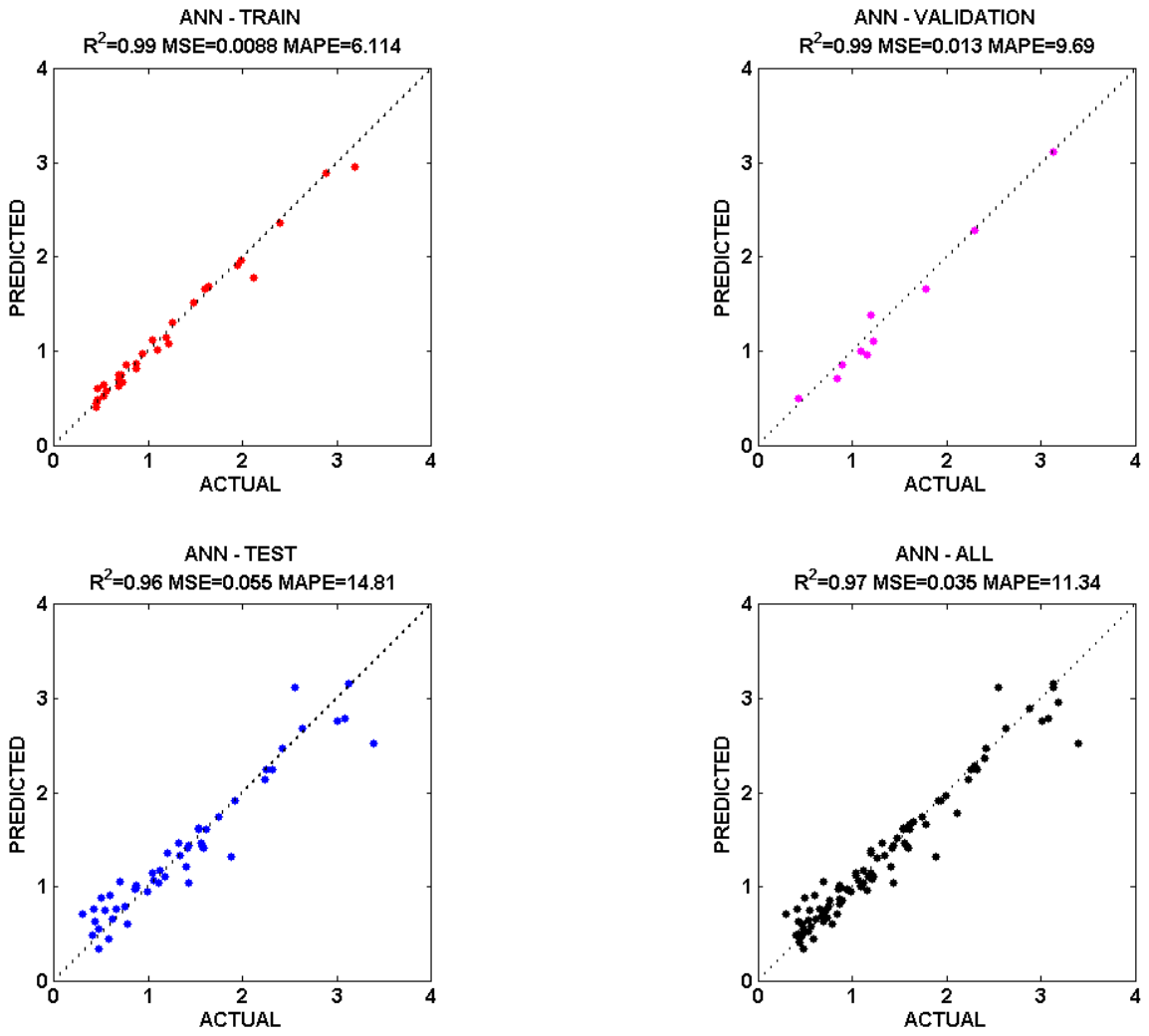


Figure 3.20 Performance of ANN model for the first batch of 40 learning points

Table 3.3 Performance summary of ANN with increasing number of training points

Number of Learning Points	Training			Validation			Testing		
	R^2	MSE	MAPE	R^2	MSE	MAPE	R^2	MSE	MAPE
40	0.99	0.0088	6.114	0.99	0.013	9.69	0.96	0.055	14.81
80	1	0.0011	2.383	1	0.0048	7.008	1	0.0057	5.948
120	1	0.00041	1.975	1	0.0021	3.697	1	0.0026	3.96
160	1	0.00019	1.455	1	0.0014	3.1	1	0.0023	2.899
200	1	0.00055	2.309	1	0.0022	3.307	1	0.0023	3.417

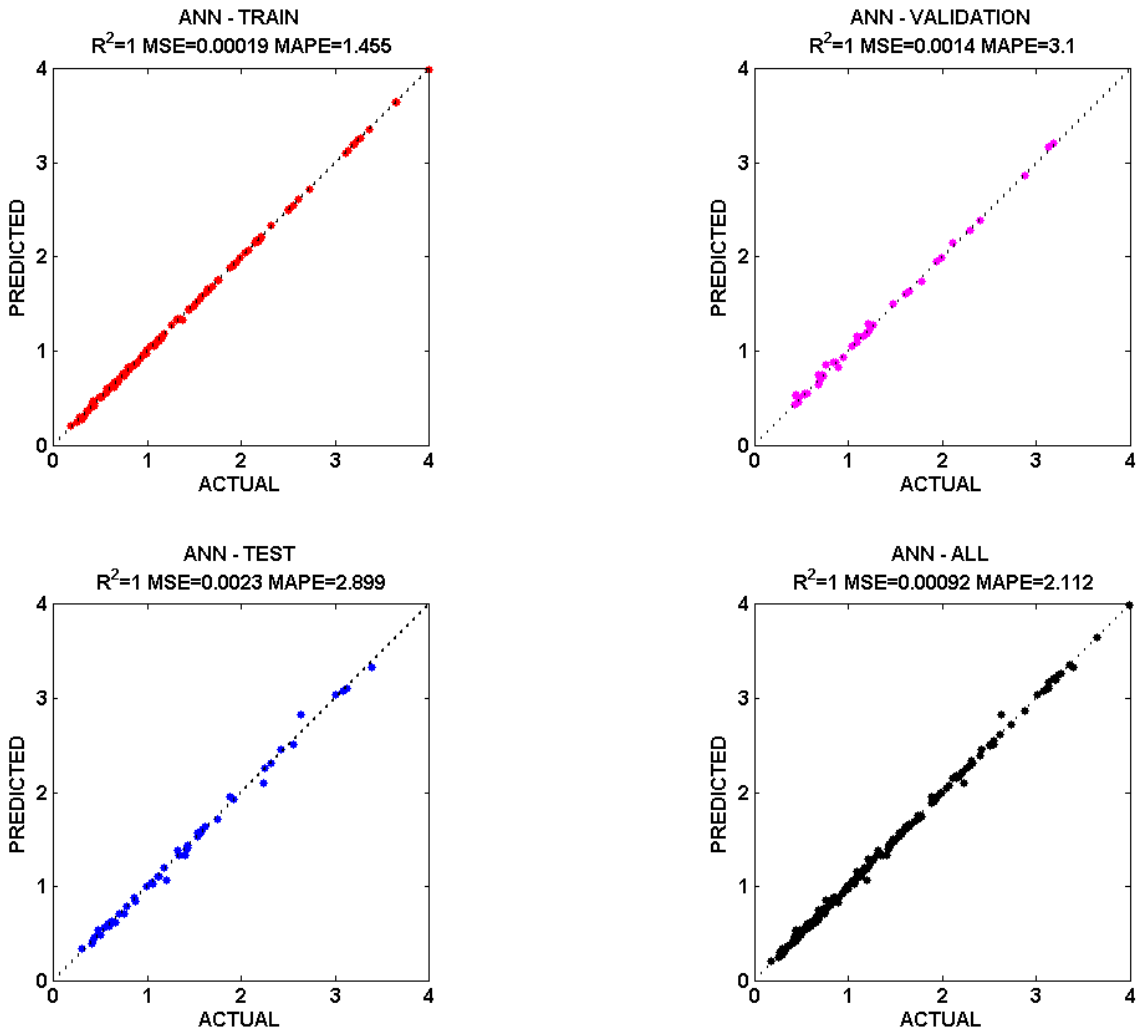


Figure 3.21 Performance of ANN model for 160 learning points

3.5.3.3 Support Vector Machines Approach

The support vector machines can be used to approximate the functional relationship between the 5 input design variables $(\boldsymbol{\beta}, \boldsymbol{H}, \boldsymbol{\phi}', \boldsymbol{c}', \boldsymbol{d})$ and the static factor of safety (F.S.). ε-SV regression (Vapnik, 1995) is used for the function approximation problem. The computer code LIBSVM Version 3.0 for MATLAB® is used (Chang and Lin, 2001). Radial basis function (RBF) is used as a kernel. Learning and testing point inputs are scaled so that they fall in the range [0, 1]. There are three parameters needed to be assigned before the training for ε-SV regression: a penalty parameter C, a deviation

parameter ε, and the RBF parameter $\boldsymbol{\gamma}$. A trial and error based parameter search is conducted to determine the optimum parameters that give the best prediction accuracy. Various combinations of C, ε and $\boldsymbol{\gamma}$ are tried in a systematic way using a grid search approach. Each parameter range is divided into a certain number of intervals and training is performed for each combination of these parameters. Using exponentially growing sequences of C, ε and $\boldsymbol{\gamma}$ (for example: $\boldsymbol{C = 2^{-5}, 2^{-3}, \cdots, 2^{15}}$, $\boldsymbol{\varepsilon = 2^{-10}, 2^{-9}, \cdots, 2^{-3}}$ and $\boldsymbol{\gamma = 2^{-15}, 2^{-13}, \cdots, 2^{3}}$) is shown to be a practical method for the initial coarse gird search (Chang and Lin, 2001). After determining the region on the three dimensional space that gives better prediction accuracy, one of the parameters is fixed and a finer grid search is performed on that region.

Similar to ANN models, the first batch of 40 learning points is used to train (30 training points + 10 validation points) the SVM model and 50 test points are used to evaluate the performance of the model. An initial coarse grid search is performed with the three parameters. It was observed that equal prediction accuracies which are higher than the rest are obtained for values of C greater or equal than $\boldsymbol{2^8}$ for the same values of ε and $\boldsymbol{\gamma}$. Therefore, the value of C is fixed at $\boldsymbol{2^8}$ and a two dimensional grid search is performed on ε and $\boldsymbol{\gamma}$. First a coarse grid search is conducted as seen in Figure 3.22. The maximum prediction accuracy, corresponding to minimum validation MSE = 0.1620 was obtained at $\boldsymbol{\gamma = 0.03125 \text{ and } \varepsilon = 0.0078175}$. Next a finer grid search is performed as seen in Figure 3.23 and better prediction accuracy (MSE = 0.1412) is obtained at $\boldsymbol{\gamma = 0.037 \text{ and } \varepsilon = 0.002}$. The performance of the SVM model for the first 40 learning points can be seen in Figure 3.24. SVM model fits the training data very well (MSE = 0.0069) but struggles with the validation data (MSE = 0.14). The performance on the test data (MSE = 0.055) is very similar to RSM model (MSE=0.058) and the ANN model (MSE = 0.055).

Subsequently SVM regression is performed on the models with more learning data and the performance of the model is evaluated with increasing number of learning points. The same grid search procedure to determine the best combination of C, ε and $\boldsymbol{\gamma}$ for the SVM regression is followed for all of the models. The performance plots of SVM models for each increment of increasing numbers of learning points can be seen in Appendix A (Figure A.7-A.9). A summary of the performances of SVM models for increasing numbers of learning points can be seen in Table 3.4.

As it can be seen in Table 3.4, the performance of the SVM regression model initially improves with adding more learning points. However, after 120 learning points performance stabilizes and is not affected by addition of more learning points. The performance of the SVM model for 120 learning points can be seen in Figure 3.25.

Table 3.4 Performance summary of SVM with different number of learning points

Number of Learning Points	Training			Validation			Testing		
	R^2	MSE	MAPE	R^2	MSE	MAPE	R^2	MSE	MAPE
40	0.99	0.0069	3.125	0.90	0.14	17.83	0.96	0.055	14.46
80	1	0.0004	1.69	0.97	0.028	11.16	0.98	0.033	12.25
120	1	0.0024	3.72	0.99	0.012	8.003	0.99	0.012	7.637
160	1	0.006	4.862	0.99	0.01	6.273	0.99	0.015	9.209
200	1	0.0051	5.802	0.99	0.013	6.997	0.99	0.016	8.145

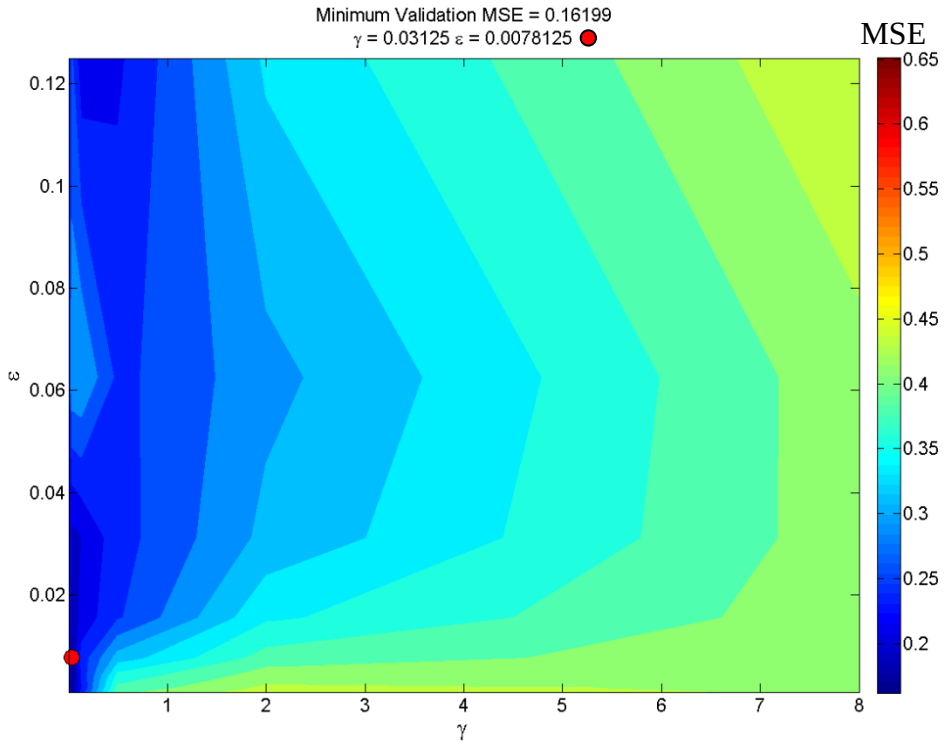


Figure 3.22 Coarse grid search for parameters ε and $\boldsymbol{\gamma}$

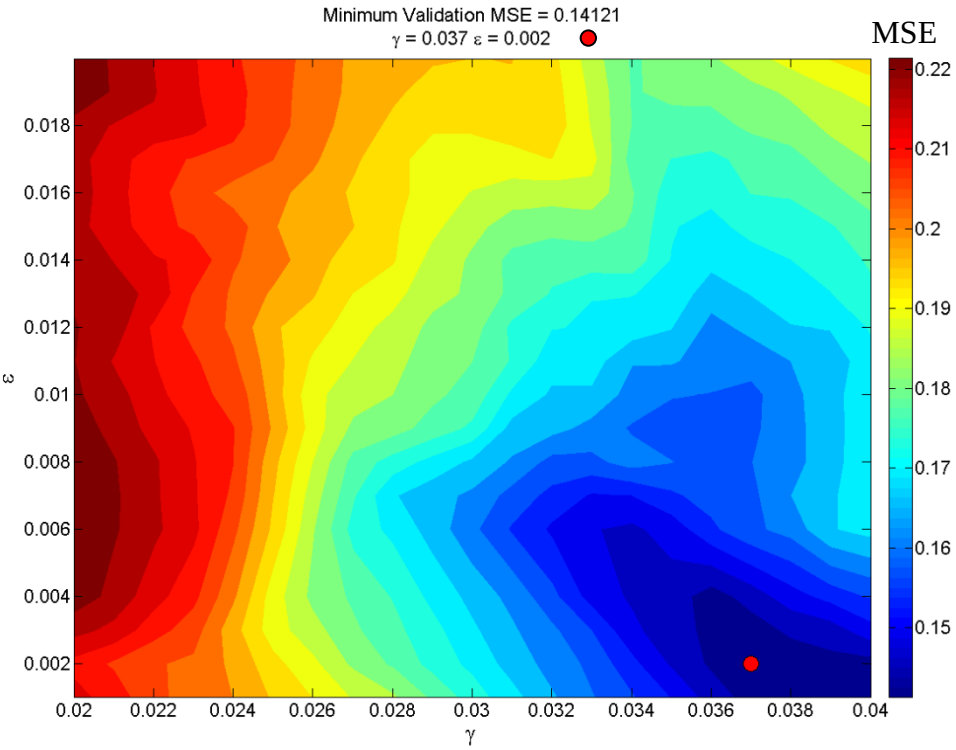


Figure 3.23 Fine grid search for parameters ε and $\boldsymbol{\gamma}$

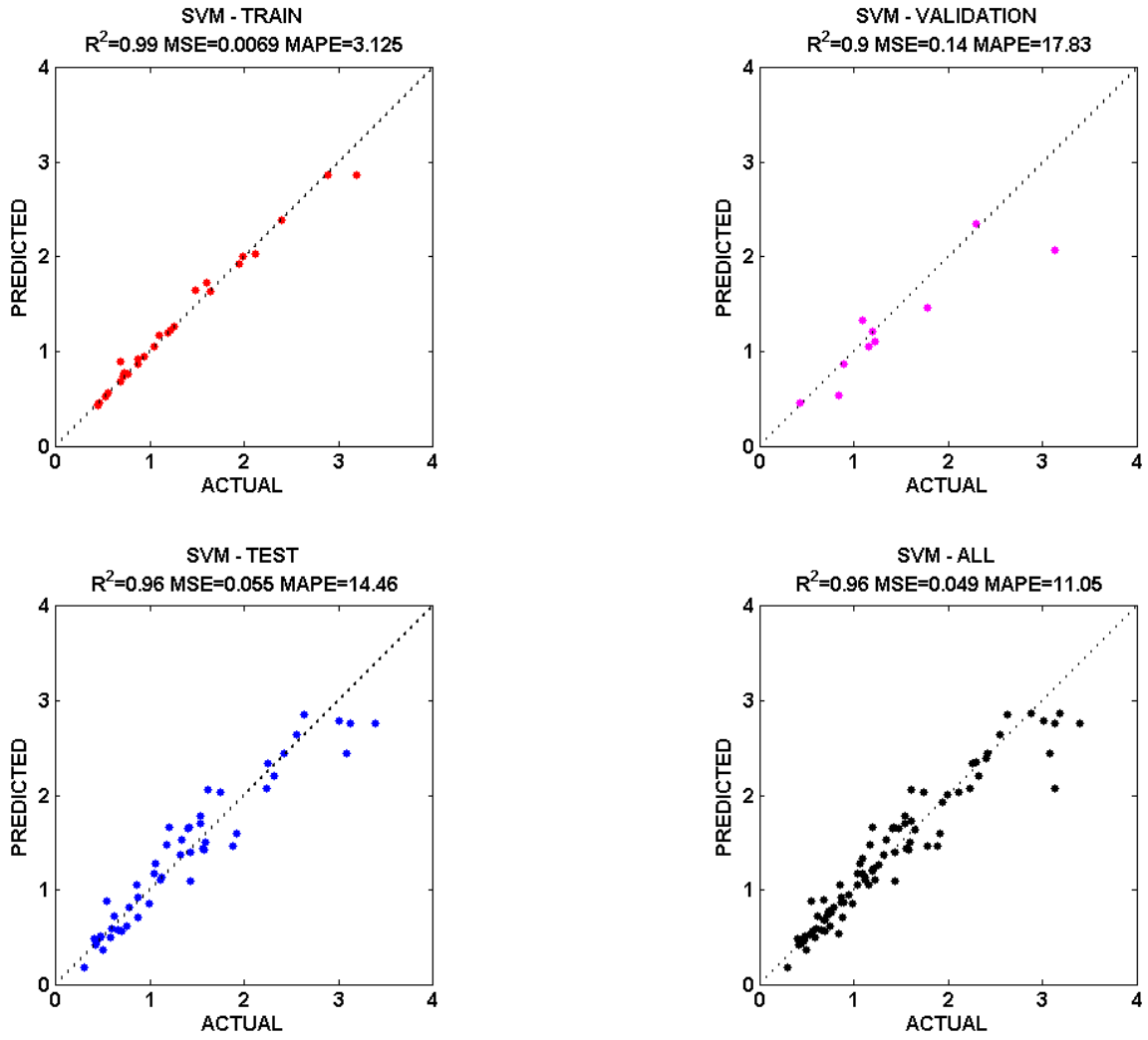


Figure 3.24 Performance of SVM model for the first batch of 40 learning points

Table 3.5 shows the performance summary of three models on the test data with increasing number of learning points. The initial performances with the first batch of 40 learning points for all three models are similar. However, with increasing number of learning points ANN model outperforms both SVM and RSM models. SVM model performs better than RSM model with increasing number of learning points.

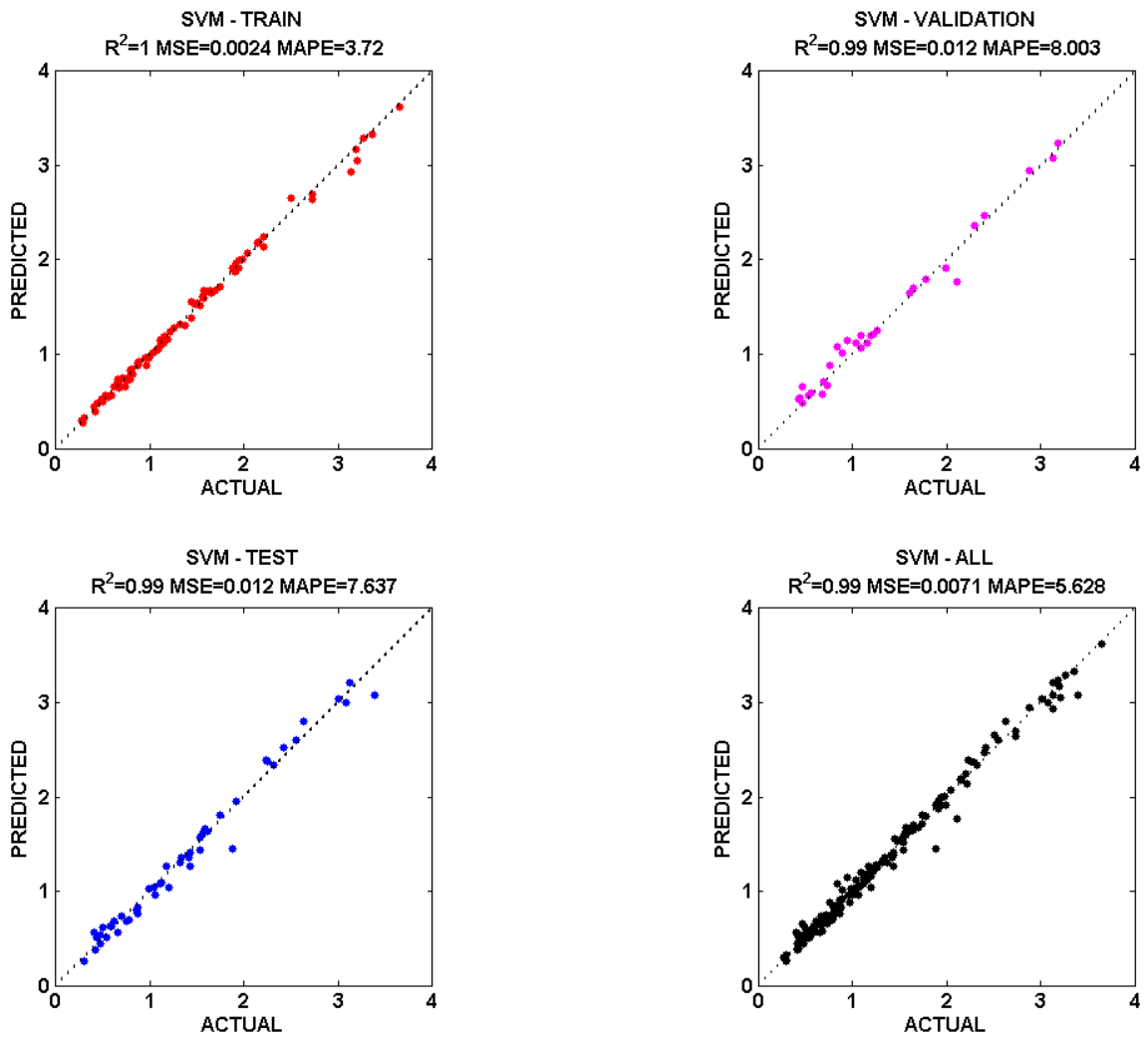


Figure 3.25 Performance of SVM model for 120 learning points

Table 3.5 Performance summary of three different models

Training Points	**RSM**			**ANN**			**SVM**		
	R^2	MSE	MAPE	R^2	MSE	MAPE	R^2	MSE	MAPE
40	0.96	0.058	35.94	0.96	0.055	14.81	0.96	0.055	14.46
80	0.99	0.019	8.998	1	0.0057	5.948	0.98	0.033	12.25
120	0.98	0.023	10.29	1	0.0026	3.96	0.99	0.012	7.637
160	0.98	0.025	9.856	1	0.0023	2.899	0.99	0.015	9.209
200	0.98	0.022	9.733	1	0.0023	3.417	0.99	0.016	8.145

Table 3.6 shows the regression results of three models for 50 test points compared with the actual factor of safety values calculated from the computer simulations. For all three models, the regression results with the best MSE performance on the test data are shown on Table 3.6.

Table 3.6 Regression results of three different models for 50 test experiments

	β	**H**	**φ'**	**c'**	**d**	**Actual F.S.**	**RSM F.S.**	**ANN F.S.**	**SVM F.S.**
1	46	282	44	21046	1753	0.75	0.62	0.73	0.80
2	53	150	46	12341	1949	0.48	0.43	0.50	0.51
3	48	57	24	7517	1648	0.50	0.56	0.52	0.67
4	56	69	48	44816	1976	1.20	1.24	1.17	1.14
5	56	311	48	36093	1778	0.48	0.45	0.51	0.49
6	19	245	46	29304	2334	3.13	3.10	3.13	3.15
7	40	105	32	6133	1623	0.70	0.64	0.71	0.69
8	12	78	21	35414	2208	2.23	2.26	2.24	2.42
9	41	21	32	26820	2430	1.43	1.13	1.32	1.28
10	25	115	26	46480	2353	1.34	1.55	1.34	1.49
11	37	19	29	49595	2493	1.89	1.56	1.91	1.79
12	31	116	36	29447	1984	1.42	1.48	1.42	1.34
13	33	154	47	10703	2090	1.54	1.55	1.58	1.46
14	28	41	43	39772	1629	2.63	2.63	2.71	2.83
15	34	53	34	42861	1903	1.61	1.70	1.61	1.59
16	52	143	30	6546	1664	0.30	0.37	0.31	0.35
17	20	203	37	44356	1923	2.32	2.42	2.31	2.35
18	46	133	42	25115	1727	0.86	0.85	0.88	0.94
19	47	308	31	22228	2250	0.44	0.43	0.45	0.49
20	44	252	38	46048	2274	0.78	0.73	0.78	0.73
21	17	262	24	33876	2181	1.59	1.74	1.60	1.71
22	12	27	30	19862	2282	3.40	2.98	3.26	3.07

Table 3.6 continued

	β	H	φ’	c’	d	Actual F.S.	RSM F.S.	ANN F.S.	SVM F.S.
23	36	81	39	37109	2417	1.40	1.51	1.38	1.43
24	42	206	35	8211	1958	0.66	0.50	0.64	0.59
25	21	226	48	33717	1689	3.08	2.96	3.08	3.25
26	15	146	37	22965	2392	3.01	2.96	2.97	2.89
27	39	85	35	9372	2237	0.87	0.81	0.85	0.77
28	23	325	32	36882	2202	1.56	1.76	1.56	1.62
29	50	191	31	43513	2026	0.54	0.65	0.56	0.50
30	27	301	43	36239	1745	1.92	1.97	1.91	1.99
31	43	95	47	27785	2289	1.17	1.21	1.19	1.28
32	31	264	40	30720	1969	1.44	1.45	1.45	1.40
33	52	119	32	13278	1768	0.42	0.50	0.45	0.53
34	47	121	26	44184	2383	0.62	0.76	0.60	0.66
35	28	41	30	10099	1929	1.32	1.35	1.36	1.36
36	29	169	29	19738	1645	1.12	1.10	1.10	1.07
37	26	232	23	42100	1938	1.04	1.06	1.02	1.08
38	45	288	34	26019	1702	0.60	0.54	0.63	0.68
39	43	338	44	23104	1861	0.88	0.81	0.84	0.84
40	20	76	35	24023	1919	2.26	2.39	2.26	2.39
41	30	324	32	31143	2257	1.12	1.22	1.11	1.06
42	22	126	42	24752	1909	2.42	2.53	2.44	2.51
43	59	158	29	47211	2472	0.41	0.82	0.40	0.33
44	46	349	34	44682	2231	0.58	0.56	0.60	0.58
45	60	59	48	29892	1613	1.06	1.09	1.06	1.34
46	17	310	25	13473	1799	1.58	1.81	1.57	1.69
47	38	166	42	5680	1735	0.99	0.90	1.00	0.92
48	21	261	44	14851	2454	2.55	2.74	2.52	2.55
49	31	88	39	33207	1679	1.75	1.83	1.68	1.80
50	30	195	37	48808	1829	1.54	1.63	1.52	1.38

3.5.4 Static Factor of Safety as a Classification Problem

Other than modeling the static factor of safety as a regression problem, it can also be modeled as a classification problem where the status of each slope is defined either as stable or unstable depending on the factor of safety calculated from the computer simulations. The same set of learning and testing experiments created and simulated for the regression problem is used for the classification problem. Slopes with factor of safeties smaller than or equal to 1.0 are defined as unstable (U) and slopes with factor of safeties larger than 1 are defined as stable (S). Therefore, the problem becomes a two-class classification {U, S} with five input design variables ($\boldsymbol{\beta}, \boldsymbol{H}, \boldsymbol{\phi'}, \boldsymbol{c'}, \boldsymbol{d}$). Naïve Bayes classifiers, artificial neural networks (ANN) and support vector machines (SVM) are used to predict the status of the slopes. All of the three classifiers are trained and tested with the same sets of learning and testing data, therefore the performances of the different classifiers can be compared. Similar to the regression problem, the performances of the classifiers are examined while increasing the number of learning points.

3.5.4.1 Naïve Bayes Approach

Naïve Bayes classifier is a commonly used probabilistic classifier based on Bayes' theorem. It makes the assumption that the decision problem is posed in probabilistic terms, and that all of the relevant probabilities are known. It has strong independence assumptions such that the presence of a particular feature of a class is unrelated to the presence of any other feature however it appears to work well in practice even for cases where the independence assumption is not valid (Duda et al., 2001). Probabilities are determined from the training data and the classifier determines the posterior probability of an unseen sample belonging to each class and then the method classifies the sample according to the largest posterior probability. A detailed description of the Naïve Bayes method can be found in Duda et al. (2001).

MATLAB®, Statistics Toolbox™ is used in this study for Naïve Bayes classification. A Naïve Bayes classifier with normal (Gaussian) distribution is created and trained using the 'NaiveBayes.fit' function. Afterwards the performance of the classifier is evaluated using the test data. Since, there are no parameters to calibrate (unlike SVM and ANN); a validation set is not needed. All of the learning set is used to fit the Naïve Bayes classifier. The performance of the classifier is calculated using the percentage of correctly classified samples (see Eq. (3.36)). A confusion matrix is used to represent the performance in a more effective manner. Each row of the confusion matrix represents the instances in an actual class and each column of the matrix represents the instances in a predicted class. Using a confusion matrix makes it is easier to see which classes are mislabeled as another.

Similar to regression models, the first batch of 40 learning points is used to train the Naïve Bayes classifier and 50 test points are used to evaluate the performance of the classifier. The prediction accuracy for the training set is 90% and for the testing set is 82%. The confusion matrices for 40 training points and 50 testing points can be seen in Table 3.7.

Table 3.7 Confusion matrices of Naïve Bayes classifier for 40 training and 50 test points

		PREDICTED	
		U	S
ACTUAL	U	17	2
	S	2	19
		90%	

Training

		PREDICTED	
		U	S
ACTUAL	U	16	3
	S	6	25
		82%	

Testing

Next, Naïve Bayes classifier is trained with more training data and the performance of the classifier is evaluated with increasing number of training points. The

confusion matrices of classifiers for increasing numbers of training points can be seen in Appendix A (Table A.1-A.3). A summary of the performances of Naïve Bayes classifiers with increasing numbers of training points can be seen in Table 3.8.

Table 3.8 Performance summary of Naïve Bayes Classifier with increasing number of training points

Number of Training Points	Training	Testing
	Prediction Accuracy (%)	Prediction Accuracy (%)
40	90	82
80	82.50	86
120	85.83	88
160	88.13	88
200	88.50	90

As can be seen from Table 3.8, the prediction accuracy of the Naïve Bayes classifier initially increases with increasing the number of training points. After obtaining the prediction accuracy (88%) at 120 training points, the performance doesn't improve very much. The maximum prediction accuracy is obtained at 200 points (90%). The confusion matrices for 200 training points and 50 testing points can be seen in Table 3.9.

Table 3.9 Confusion matrices of Naïve Bayes classifier for 200 training and 50 test points

PREDICTED

ACTUAL		U	S
	U	82	10
	S	13	95
		88.5%	

Training

PREDICTED

ACTUAL		U	S
	U	17	2
	S	3	28
		90%	

Testing

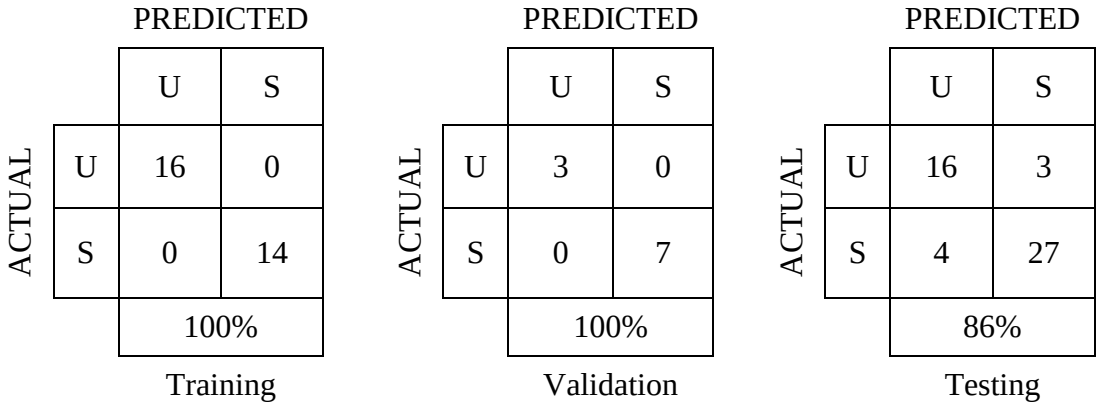
3.5.4.2 Artificial Neural Networks Approach

ANN classifier is used to model the slope stability as a two-class classification problem. MATLAB®, Neural Network Toolbox™ 'newpr' function is employed to create three layer pattern recognition networks. The input layer consists of the 5 design variables ($\boldsymbol{\beta}, \boldsymbol{H}, \boldsymbol{\phi}', \boldsymbol{c}', \boldsymbol{d}$) . The second layer is the hidden layer. The number of the neurons (N) in the hidden layer is decided by a trial and error process. The third layer is the output layer which consists of two classes; stable (S) and unstable (U). The network is trained with the learning data set and the performance of the network is evaluated by the use of the test data set. The prediction accuracy, Eq. (3.36) is used as the performance measure. A sigmoid transfer function is used for the hidden layer and a linear transfer function is used for the output layer. Prior to training, input design variables are scaled so that they fall in the range [-1, 1].

The first batch of 40 learning points (30 training points + 10 validation points) is used to train the ANN classifier and 50 test points are used to evaluate the performance of the classifier. 10 neurons are used in the hidden layer. The prediction accuracy for the training set is 96.7%, for the validation set it is 100% and for the testing set it is 86%. The performance of the ANN classifier on test data is slightly better than the performance of the Naïve Bayes classifier (82%) for the first batch of 40 learning points. The confusion matrices for 40 learning points and 50 testing points can be seen in Table 3.10.

ANN classifier is trained with more learning data and the performance of the classifier is evaluated with increasing number of learning points. The confusion matrices of classifiers for different numbers of learning points can be seen in Appendix A (Table A.4-A.6). A summary of the performances of ANN classifiers with increasing numbers of learning points can be seen in Table 3.11.

Table 3.10 Confusion matrices of ANN classifier for 40 learning and 50 test points

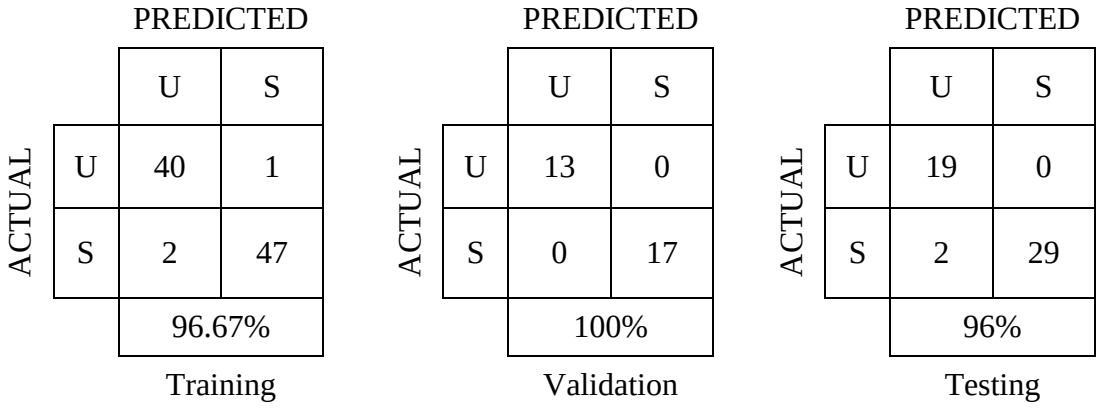


As it can be seen in Table 3.11, the prediction accuracy of the ANN classifier improves initially by increasing the number of learning points. After 120 learning points the prediction accuracy is 96% and it reaches a maximum. The confusion matrices for 120 learning points and 50 testing points can be seen in Table 3.12.

Table 3.11 Performance summary of ANN Classifier with increasing number of learning points

Number of Learning Points	Training	Validation	Testing
	Prediction Accuracy (%)	Prediction Accuracy (%)	Prediction Accuracy (%)
40	100	100	86
80	96.67	100	92
120	96.67	100	96
160	96.67	100	96
200	99.33	98	94

Table 3.12 Confusion matrices of ANN classifier for 120 learning and 50 test points



3.5.4.3 Support Vector Machines Approach

Slope classification problem is modeled using support vector machine (SVM) classifier. LIBSVM Version 3.0 for MATLAB® (Chang and Lin, 2001) is used for the classification problem. Radial basis function (RBF) is used as a kernel. There are two parameters needed to be assigned before the training for SVM classifier: a penalty parameter *C* and the RBF parameter $\boldsymbol{\gamma}$**.** Grid search method explained in section 3.5.3.3 is used to determine the optimum values of *C* and $\boldsymbol{\gamma}$ that give the best prediction accuracy. These parameters are calibrated using cross-validation on the learning data. Therefore, unlike ANN classifier, there is no validation set for SVM classifier.

Table 3.13 shows the confusion matrices of the SVM classifier for 40 learning data and 50 testing data. The training prediction accuracy is 100% and the testing prediction accuracy is 92%. SVM classifier performs better than both ANN and Naïve Bayes classifiers for the first batch of 40 learning data.

Table 3.13 Confusion matrices of SVM classifier for 40 learning and 50 test points

		PREDICTED	
		U	S
ACTUAL	U	19	0
	S	0	21
		100%	

Training

		PREDICTED	
		U	S
ACTUAL	U	16	3
	S	1	30
		92%	

Testing

SVM classifier is trained with more learning data and the performance of the classifier is evaluated with increasing number of learning points. The confusion matrices of classifiers for different numbers of learning points can be seen in Appendix A (Table A.7-A.9). A summary of the performances of SVM classifiers with increasing numbers of

learning points can be seen in Table 3.14. Unlike Naïve Bayes and ANN classifiers, SVM classifier starts with high percentage prediction accuracy (92%) and the prediction accuracy reaches 100% with 80 learning points. While the number of learning points increase, the prediction accuracy is stable with small fluctuations. The confusion matrices for 80 learning points and 50 testing points can be seen in Table 3.15.

Table 3.14 Performance summary of SVM Classifier with different number of training points

Number of Learning Points	Training	Testing
	Prediction Accuracy (%)	Prediction Accuracy (%)
40	100	92
80	100	100
120	95.8	100
160	100	98
200	98.5	100

Table 3.15 Confusion matrices of SVM classifier for 80 learning and 50 test points

		PREDICTED	
		U	S
ACTUAL	U	36	0
	S	0	44
		100%	

Training

		PREDICTED	
		U	S
ACTUAL	U	19	0
	S	0	31
		100%	

Testing

Table 3.16 shows the final comparison of the performances of three different classifiers with different number of learning points. With the minimum number of training points SVM classifier shows the best performance (92%). ANN classifier (86%) follows the SVM classifier and performs better than the Naïve Bayes classifier (82%). SVM classifier clearly outperforms both ANN and Naïve Bayes classifiers at all number of training points. ANN classifier outperforms the Naïve Bayes classifier at every number of learning points.

Table 3.16 Testing performance summary of three different classifiers

Number of Learning Points	**Prediction Accuracy (%)**		
	Naïve Bayes	ANN	SVM
40	82	86	92
80	86	92	100
120	88	96	100
160	88	96	98
200	90	94	100

3.5.5 *k*-fold Cross-validation of Different Approaches

Stratified repeated *k*-fold cross-validations are used to compare the performances of different regression and classification models in a less biased manner. Since 250 experimental points (200 learning set + 50 test set) are already created to evaluate the performance of the models with increasing number of learning points, the same set of experiments can be used for stratified repeated *k*-fold cross-validation. Other than using just one testing set and an increasing number of learning data set, stratified repeated *k*-fold cross-validation is performed on all available data. The cross-validation is performed for increasing number of experimental points starting from 40 experimental points and ending at 240 experimental points in increments of 40 points. Therefore, the effect of sample size on the performance can also be investigated. The number of folds is chosen

as 10 which is the most commonly used number of folds (Kim, 2009). 10-fold cross-validation is performed on each model including both regression and classification models. In the stratified 10-fold cross-validation, the whole set is divided into 10 subsets of approximately equal size. In stratified cross-validation, the folds are created such that the mean output of each fold is the same as the initial set for the regression problems and each fold contains roughly the same proportions of the type of classes for the classification problems. Each fold is then used as a test set with the model trained with the remaining 9 folds. The training is repeated 10 times with each separate fold. The performance of each model is then calculated by averaging the performance measures from 10 separate training-testing steps. For regression models MSE is chosen the performance measure and for classification models accuracy percentage is chosen as the performance measure. In order to reduce the variation due to randomness of partitioning the sample into *k*-folds, stratified *k*-fold cross validation is repeated 10 times and the average is taken as the performance measure.

Table 3.17 shows the stratified repeated 10-fold cross-validation results for RSM, ANN and SVM regression models. For each model, the same folds and repetitions are used so that the performances of the models can be compared. Minimum, maximum and average mean square errors (MSE) of the repetitions are shown for each model with respect to the number of points used. In accordance with the previous performance results with fixed test data set, in the cross-validation assessment ANN regression model outperforms both RSM and SVM models. SVM model performs considerably better than the RSM model. The cross-validation performances of all of the models initially increase with increasing number of points and then stabilize after certain number of points. The stratified repeated 10-fold cross-validation is also performed on the classification models. The same folds that were used in the regression models are used for the classification models.

Table 3.17 Stratified repeated 10-fold cross-validation results for regression models

Number of Points	RSM			ANN			SVM		
	Min. MSE	Max. MSE	Ave. MSE	Min. MSE	Max. MSE	Ave. MSE	Min. MSE	Max. MSE	Ave. MSE
40	0.0434	0.0440	0.0437	0.0676	0.1766	0.1265	0.0282	0.0286	0.0284
80	0.0380	0.0403	0.0395	0.0133	0.0328	0.0176	0.0281	0.0309	0.0291
120	0.0263	0.0277	0.0268	0.0037	0.0099	0.0066	0.0164	0.0174	0.0169
160	0.0275	0.0309	0.0288	0.0024	0.0117	0.0043	0.0110	0.0133	0.0121
200	0.0236	0.0243	0.0239	0.0019	0.0027	0.0022	0.0065	0.0076	0.0072
240	0.0233	0.0244	0.0237	0.0015	0.0023	0.0018	0.0053	0.0067	0.0062

The results of the cross-validation can be seen in Table 3.18. Minimum, maximum and average prediction accuracies of repetitions are shown for each model with respect to the number of points used. Average prediction accuracy from all 10 repetitions is used as the performance measure of each model. In cross-validation performance evaluation of different classifiers, SVM classifier outperformed both ANN classifier and Naïve Bayes classifier at all number of points.

Table 3.18 Stratified repeated 10-fold cross-validation results for classification models

	Prediction Accuracy (%)								
Number of Points	NB			ANN			SVM		
	Min.	Max.	Ave.	Min.	Max.	Ave.	Min.	Max.	Ave.
40	85.00	87.50	86.50	77.50	87.50	83.25	92.50	95.00	93.75
80	80.00	83.75	81.75	82.50	91.25	86.88	90.00	92.50	91.75
120	80.00	85.00	82.83	83.33	92.50	88.58	90.83	95.00	92.75
160	85.00	88.13	86.44	90.63	95.00	92.19	93.75	97.50	95.69
200	85.50	87.50	86.55	93.50	96.50	94.80	95.50	97.00	96.20
240	85.83	87.50	86.54	93.75	96.25	94.79	95.83	97.50	96.83

3.6. Summary

The concept of building metamodels based on the data obtained from computationally expensive computer simulations is introduced. A feasibility study is performed and different metamodeling techniques are compared. From the results of the feasibility study, it can be concluded that for regression problem in terms of prediction accuracy ANN based metamodels performed noticeably better than both RSM and SVM based metamodels. However, for the classification problem SVM outperformed both ANN and Naïve Bayes classifier. On both classification and regression models, more complicated machine learning algorithms ANN and SVM performed better than relatively simplistic RSM and Naïve Bayes models.

For regression models, in terms of computational time required to create, train and test the metamodel, RSM is considerably faster than both ANN and SVM. RSM doesn't require any parameter calibration therefore even with higher number of input variables and experimental points, it takes less than 30 seconds to train and test the RSM metamodel on a 2.8 GHz Intel® Core™ i7 computer. ANN and SVM regressors require the calibration of the parameters. The computational time required for building ANN and SVM models, increases with increasing number of experimental points. For the study presented in this chapter training/validating ANN model with 200 points and testing with 50 points take about 2 minutes on the same computer. For the same number of experimental points SVM takes about 4 minutes because SVM regression model requires calibration of 3 different parameters and a grid search on a 3-D space is time consuming.

For classification models, Naïve Bayes classifier is noticeably faster than both ANN and SVM. Naïve Bayes classifier also doesn't require any parameter calibration. It takes only seconds to train and test the Naïve Bayes metamodel. ANN and SVM classifiers require the calibration of the parameters. Training/validating both ANN and SVM models with 200 points and testing with 50 points take about 2 minutes.

Computational time required for SVM classifier is less than SVM regressor because the classifier requires only calibration of 2 different parameters.

CHAPTER 4

PREDICTING SEISMIC RESPONSES OF SLOPES USING SOFT COMPUTING METHODS

4.1 Seismic Slope Stability Analysis Using Finite Difference Method

FLAC (Fast Lagrangian Analysis of Continua) is a two-dimensional explicit finite difference program for engineering mechanics computation. FLAC uses a fully nonlinear analysis method which is based on the explicit finite difference scheme to solve full equations of motions. Unlike equivalent-linear methods, only one run is performed with a fully nonlinear method, because nonlinearity in the stress-strain law is followed directly by each element as the solution marches in time (Itasca, 2008). If an appropriate nonlinear law can be defined, the dependence of damping and apparent modulus on strain level are automatically modeled. The nonlinear numerical method has not been applied as frequently in practical design. However, applications using nonlinear numerical codes are becoming increasingly popular due to the fact that more importance is placed on making reliable prediction of permanent deformations in geostructures.

In order to provide accurate wave propagation, the spatial element size, Δl, must be smaller than approximately one-tenth of the wavelength associated with the highest frequency component of the input wave. Therefore, the element sizes are determined accordingly. A rigid base boundary condition is assumed and the seismic input is applied at the bottom boundary of the model as an acceleration time history. The boundary conditions at the side of the model should minimize the wave reflections and achieve free-field conditions. Therefore, free-field boundary conditions are implemented on the lateral boundaries of the model using viscous dashpots (Itasca, 2008). The material constitutive model is assumed as elastic for the initial gravity stress analysis and Mohr-Coulomb plasticity model is used for the dynamic analysis. A non-associative flow rule

(dilation angle, $\psi = 0°$) is used. A full Rayleigh damping scheme (stiffness and mass-proportional components) is used to model the material damping. Rayleigh damping ($\xi = 2.5\%$) is applied at the mean frequency of the input ground motion. In dynamic analyses that use Mohr-Coulomb plasticity constitutive models, a considerable amount of energy dissipation occurs during the plastic flow, therefore the use of a higher damping was deemed unnecessary (Itasca, 2008; Strenk, 2010).

4.2 Input Accelerations

The reliable estimation of the characteristics of ground motion that occurs during earthquakes is one of the most important parts of a regional hazard analysis, be it probabilistic or deterministic. Unlike simplified approaches that use only a number of strong motion parameters (e.g. Peak Ground Acceleration, PGA, or Arias Intensity), all finite difference/element programs including FLAC require the user to input a time-history of the ground motion. The most desirable input time-history for seismic slope stability would be a ground motion recorded during an earthquake occurred on the fault of interest and having a magnitude of adequate size (Miles and Ho, 1999). However, such records can be very difficult and most of the time impossible to obtain. In the case where only a few records are available for a region, these records should be scaled according to different site to source distances. Simple scaling of earthquake records does not properly account for the ground motion characteristics and significantly alters the frequency content of the earthquake. Also for the sequential metamodeling approach adopted in this study, there is a need to be able to create input accelerations for predetermined parameter ranges. For example, using a design of experiments approach introduced in Chapter 3, earthquake moment magnitude (M_w) and distance from the epicenter of the earthquake (R) can be used as design variables and artificial earthquake motions can be created for each of the designed experiments.

4.2.1 Methods for Generation of Artificial Earthquake Motions

There are several techniques for creating artificial ground motions for engineering purposes. Some of these procedures are based on explicit physical models of the earthquake source, travel path and recording site. However, earthquake mechanisms are so complex that physical modeling alone is currently not sufficient to create reasonable models. There is also an experimental approach where a mathematical model is fitted to the experimental data (recorded real ground motions). Moreover, there are hybrid approaches combining elements of both approaches. All of the physics-based deterministic methods convolve the source function with synthetic Green's functions to produce the motion at the ground surface (Douglas and Aochi, 2008). Most of the physics-based methods are extensively used for research purposes. The complex and computationally expensive nature of these methods prohibits their use in general engineering practice and particularly in regional seismic hazard analysis (Miles and Ho, 1999; Douglas and Aochi, 2008). An appropriate alternative to the Green's function based physical models is stochastic generation of earthquake time histories in the frequency domain. Many stochastic models successfully characterize source and site parameters and computationally efficient. There are point-source and extended source (finite-fault) stochastic models available (Brune, 1970 ; Beresnev and Atkinson, 1998; Boore, 2000; Motazedian and Atkinson, 2005). Refer to Douglas and Aochi (2008) for detailed explanation of methods for generating artificial earthquake records.

4.2.2 SMSIM for Simulating Ground Motion

SMSIM – Fortran Programs for Simulating Ground Motions from Earthquakes (Boore, 2000) is a stochastic point-source method based on the assumption that the amplitude of the ground motion at a site can be specified in a deterministic way, with a random phase spectrum modified such that the motion is distributed over a duration related to the earthquake magnitude and the distance from the source. It is particularly

useful for simulating the higher-frequency ground motions of most interest to engineers. The model uses a filtered shot noise approach that filters the windowed Gaussian white noise with theoretical Fourier amplitude in the frequency domain. SMSIM is a well established model which has been shown to successfully characterize ground motion (Boore, 2003). Miles and Ho (1999) successfully used SMSIM for generating artificial earthquake records for a regional landslide hazard zonation.

In this chapter, SMSIM is chosen for creating artificial earthquake records as an input to FLAC models because of its computational cost and abundant supporting information on its successful use in the literature.

4.3 Predicting Seismic Responses of Single Slope under Different Earthquake Loads

In this section of the thesis, the performance of a homogenous dry slope with predefined geometric and material properties is investigated under different earthquake loads which consist of both synthetic and real earthquake records. The geometric and material properties chosen for the slope can be seen in Table 4.1. The static factor of safety of this specific slope is calculated using the strength reduction method (SRM) with FLAC and the factor of safety is equal to 1.24. The Poisson ratio of the slope material is chosen as $\nu = 0.35$. The depth to bedrock from the base of the slope is chosen as 20 meters.

Table 4.1 Geometric and material properties for the single slope

H	β	d	c'	ϕ'	V_s
40 m	40^o	1900 kg/m^3	26 kPa	34^o	420 m/sec

A metamodeling approach is adopted to approximate the seismic performance of the slope as a function of earthquake loading using synthetic earthquake records. Earthquake magnitude **(Mw)** and distance from the epicenter of the earthquake **(R)** are

chosen as design variables for creating the artificial time-histories. The ranges chosen for the design variables M_w and R can be seen in Table 4.2. These ranges are used to create experimental points using a design of experiments approach.

Table 4.2 Assigned ranges for design variable values

	$\boldsymbol{M_w}$	$\boldsymbol{R}$
Lower bound	6	5 km
Upper bound	8.5	30 km

The artificial earthquake records are created using SMSIM for the given magnitude and distance combinations determined by the experimental design. SMSIM requires basic input parameters that specify the particular model for which the motions will be simulated. For the input parameters of SMSIM, a stochastic model for California is used (Atkinson and Silva, 2000). The source parameters for Atkinson and Silva (2000) model can be seen in Table 4.3. For the site amplification parameters, generic rock (Boore and Joyner, 1997) is used so the created artificial earthquake records can be applied directly at the rigid base of the FLAC model without any deconvolution analysis.

Table 4.3 Parameters for Atkinson Silva (2000) model

- $\rho_s, \beta_s, V, \langle R_{\Theta\Phi} \rangle, F, R_0$: $2.8, 3.5, 0.707, 0.55, 2.0, 1.0$
- Geometrical spreading (including factors to insure continuity of function):
  $r < 40km : 1/r$
  $40km \leq r : (1/40)(40/r)^{0.5}$
- Q, c_Q : $180f^{0.45}, 3.5\ km/s$
- Source duration: $0.5/f_a$
- Path duration: $0.05\ R$
- Site amplification: generic rock (Boore and Joyner, 1997)
- Site diminution parameters (f_{max}, κ): $100.0, 0.030$

A sequential Latin hypercube sampling approach is used for the design of experiments. First, 100 experiments are created using a *maximin* Latin hypercube design. This initial set is used only for testing the predictive accuracy of the metamodel and is not

used in the learning phase. As part of the sequential approach, 50 more experiments are augmented to the original 100 experiments using the augmented Latin hypercube design. This second set of 50 experiments is used for training and validation of the metamodels. Subsequently, the number of learning points is increased in increments of 50 experimental points up to a total of 250 experimental points. FLAC simulations are performed for each designed experiment using the same slope but different earthquake records created using SMSIM for different M_w and R combinations. The performances of the metamodels are investigated while increasing the number of learning points.

It is desirable to be able to estimate the response of a slope using different intensity measures which characterize the earthquake time history. A metamodel can be created using a vector of intensity measures as inputs. Seismic hazard analyses rely heavily on the characterization of strong ground motion by intensity measures. These intensity measures can be calculated using different ground motion prediction equations (GMPE) for a particular site and then can be used to estimate the slope response. Due to the complexity of earthquake motions, the use of a single intensity measure to characterize the ground motion is not usually sufficient. Several intensity measures are usually required to satisfactorily describe the ground motion (Kramer, 1996). Some intensity measures may represent only the amplitude of the motion; others may describe only the frequency content or the duration of the motion. There are also intensity measures that represent two or three of these characteristics.

The intensity measures are chosen such that they are readily available via one or more ground motion prediction equations (GMPE) so that regional slope failure analysis can be performed. A list of the intensity measures selected for this study can be seen in Table 4.4.

Peak quantities such as PGA and PGV only represent ground motion intensity. There are several ground motion prediction equations for these peak quantities and 5% damped pseudo-absolute- acceleration spectra (PSA) at periods between 0.01 and 10 sec

derived by empirical regression of the PEER NGA strong-motion database (Boore and Atkinson, 2007; Campbell and Bozorgnia, 2007; J. Chiou and Youngs, 2008).

Table 4.4 Intensity Measures (IM) used in the study

IM	Description
PGA	Peak Ground Acceleration
PGV	Peak Ground Velocity
I_a	Arias Intensity
CAV	Cumulative Absolute Velocity
ASI	Acceleration Spectrum Intensity
VSI	Velocity Spectrum Intensity
T_m	Mean Period
$S_a(1.5T_s)$	Spectral Acceleration at a 1.5 times fundamental site period T_s

Arias intensity (I_a) represents intensity, frequency content and duration. Arias intensity is defined as

$$I_a = \frac{\pi}{2g}\int_0^{\infty}[a(t)]^2 dt \qquad (\mathbf{4.1})$$

where $a(t)$ is the acceleration time series at time t and g is the gravitational acceleration. The Arias intensity has the unit of velocity. Arias intensity can also be predicted using several GMPE's (Kramer, 1996).

The cumulative absolute velocity (CAV) is defined as the integral of the absolute acceleration time series. There are few ground motion prediction equations for this intensity measure (Campbell and Bozorgnia, 2010). CAV can be represented mathematically by the following equation:

$$CAV = \int_0^{tmax} |a(t)|dt \qquad (\mathbf{4.2})$$

where $|a(t)|$ is the absolute value of the acceleration time series at time t and $tmax$ is the total duration of the earthquake record.

Acceleration Spectrum Intensity (ASI) is defined as the integral of the pseudospectral acceleration of a ground motion from 0.1 to 0.5 sec. ASI also can be estimated using GMPEs (Bradley, 2010). ASI is given by the equation:

$$ASI = \int_{0.1}^{0.5} S_a(T, 5\%)dT \qquad (\mathbf{4.3})$$

where $S_a(T, 5\%)$ is the 5% damped spectral acceleration.

Velocity Spectrum Intensity is defined as the in integral of the pseudospectral velocity (PSV) over the period range of 0.1 to 2.5 seconds as given by Equation. VSI can be calculated using GMPEs (Bradley et al., 2009).

$$SI = \int_{0.1}^{2.5} PSV(T, 5\%)dT \qquad (\mathbf{4.4})$$

where $PSV(T, 5\%)$ is the 5% damped pseudospectral velocity.

Mean Period (T_m) is computed from the Fourier Amplitude Spectrum and defined as

$$T_m = \frac{\sum_i C_i^2(1/f_i)}{\sum_i C_i^2} \qquad (\mathbf{4.5})$$

for $0.25\ Hz \leq f_i \leq 20\ Hz$. Where C_i is the Fourier amplitudes of the entire accelerogram and f_i is the discrete Fourier frequencies between 0.25 and 20 Hz. The Fourier amplitude is defined as the square root of the sum of squares of the real and imaginary parts of the Fourier coefficient (Rathje et al., 1998). Rathje et al. (1998) judged T_m to be the best simplified frequency content characterization parameter. Mean Period (T_m) can be estimated using GMPE's (Rathje et al., 2004).

The selected intensity measures can be classified as only earthquake record depended or both slope and earthquake record dependent. Spectral Acceleration at a 1.5 times fundamental site period T_s is both slope and earthquake record dependent and the rest is only earthquake record dependent. Initial 1D fundamental period at the crest of the slope is used as T_s and it is calculated using the following formula:

$$T_s = \frac{4H}{V_s} \qquad (\mathbf{4.6})$$

where H is the height of the crest from the bedrock and V_s is the initial shear wave velocity of the slope material.

Each one of the artificial earthquake records created using SMSIM for a given M_w and R is analyzed and selected intensity measures are calculated for each record in order to be used as inputs for the metamodels.

The displacement value calculated at the crest of the slope using FLAC is assigned as the damage index of the slope. First, the problem is considered as a regression problem and the displacement at the crest is used as the output. A multi-layer feed-forward neural network is used as the metamodel for the regression problem. The mean square error (MSE) of the predicted crest displacements is used as the performance measure of the metamodel. The ANN structure consists of three layers. The first layer is the input layer with a vector of 8 intensity measures shown in Table. The second layer is the hidden layer. The number of neurons in the hidden layer is decided by a trial and error process. The third layer is the output layer which is a vector of the crest displacements for each designed experiment. A linear transfer function is used for the output layer and a sigmoid transfer function is used for the hidden layer. Both input and outputs are scaled so that they fall in the range [-1, 1].

The neural network is trained and validated using the first batch of 50 learning points and the performance is evaluated using 100 test points. The performance of the

ANN regression model for the first 50 learning points using 10 neurons in the hidden layer can be seen in Figure 4.1. Even for a relatively low number of learning points, ANN regression performs well on the test data with MSE=0.012.

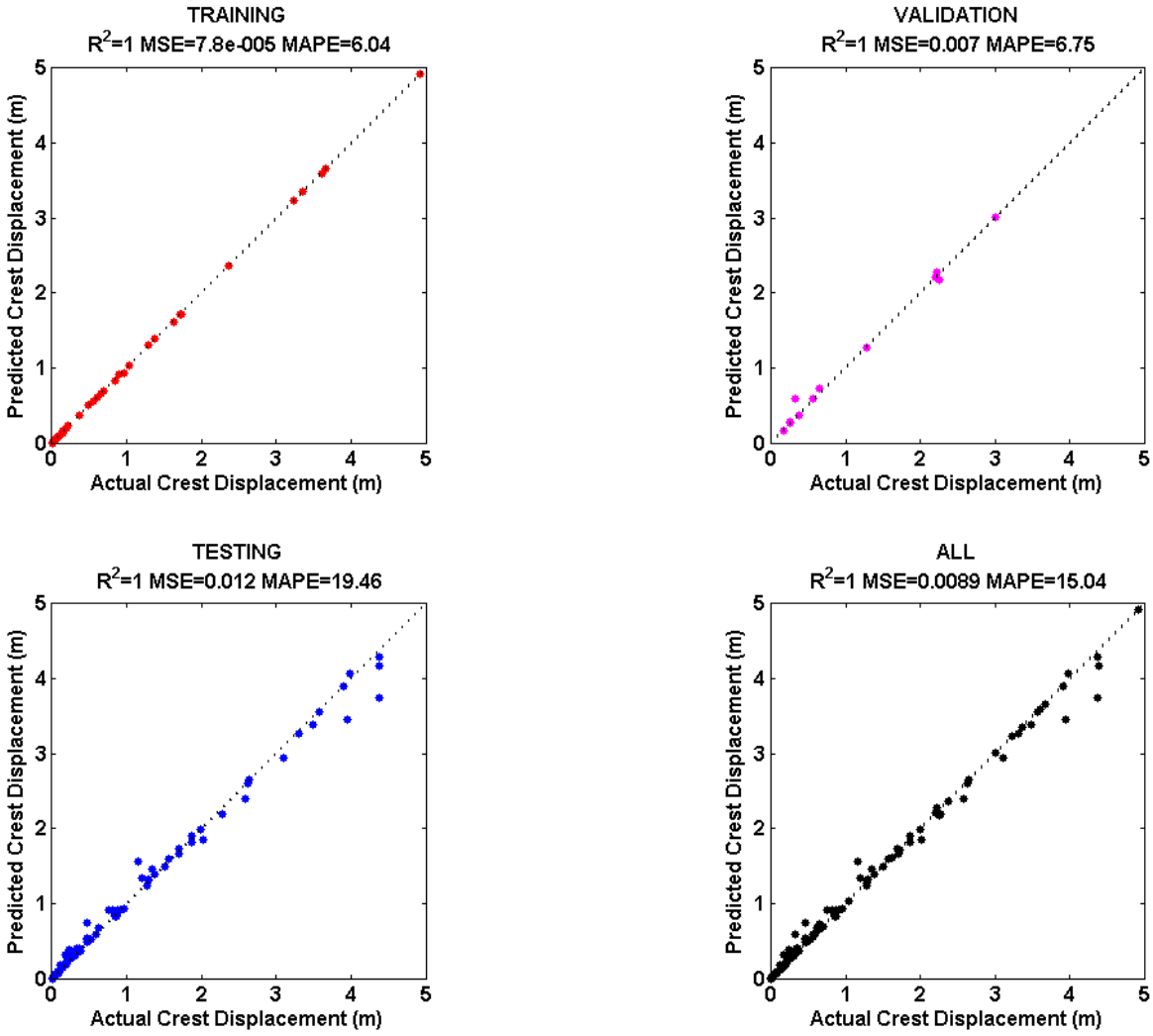


Figure 4.1 Performance of ANN model for 50 learning points for single slope case

Different neural networks are trained and validated with increasing number of learning points. The performance of the ANN model is evaluated with respect to the number of learning points. The plots of actual versus predicted values of crest displacement for increasing number of learning points can be seen in Appendix B (Figure

B.1-B.3). Table 4.5 summarizes the performance of ANN model for increasing numbers of learning points. As can be seen in the Table 4.5, the performance of the ANN model initially increases with increasing number of learning points but after 150 learning points the performance does not improve any further. The performance of the ANN regression model for 250 learning points using can be seen in Figure 4.2.

Table 4.5 Performance summary of ANN with increasing number of training points

Number of Learning Points	Training			Validation			Testing		
	R^2	MSE	MAPE	R^2	MSE	MAPE	R^2	MSE	MAPE
50	1	0.000078	6.04	1	0.007	6.75	1	0.012	19.46
100	1	0.00012	8.08	1	0.0021	8.51	1	0.0017	14.82
150	1	0.000091	11.17	1	0.00055	12.25	1	0.00092	12.81
200	1	0.00018	6.763	1	0.00054	16.17	1	0.00075	9.818
250	1	0.00023	11.04	1	0.00036	9.33	1	0.00072	14.92

Moreover, in order to get a less biased estimate of the performance of the metamodel, a 10-fold cross-validation is performed using the total 350 experimental points (100 test points + 250 learning points). The whole experiment set (350 experiments) is randomly divided into 10 subsets of equal size (35 experiments). Each subset of 35 experiments is then used as a test set with the model trained and validated with the remaining 315 experiments. The performance of the ANN model is then calculated by averaging the performance measures (MSE) from 10 separate learning-testing steps. The average MSE for the 10-fold cross-validation is 0.00064. The average MSE conforms well to the values shown in Table 4.5. Figure 4.3 shows the actual versus predicted crest displacements for each of the test set of 10-folds. As it can be seen in Figure 4.3, the 10-fold cross validation proves that the ANN model performs well with different subsets of the whole data set.

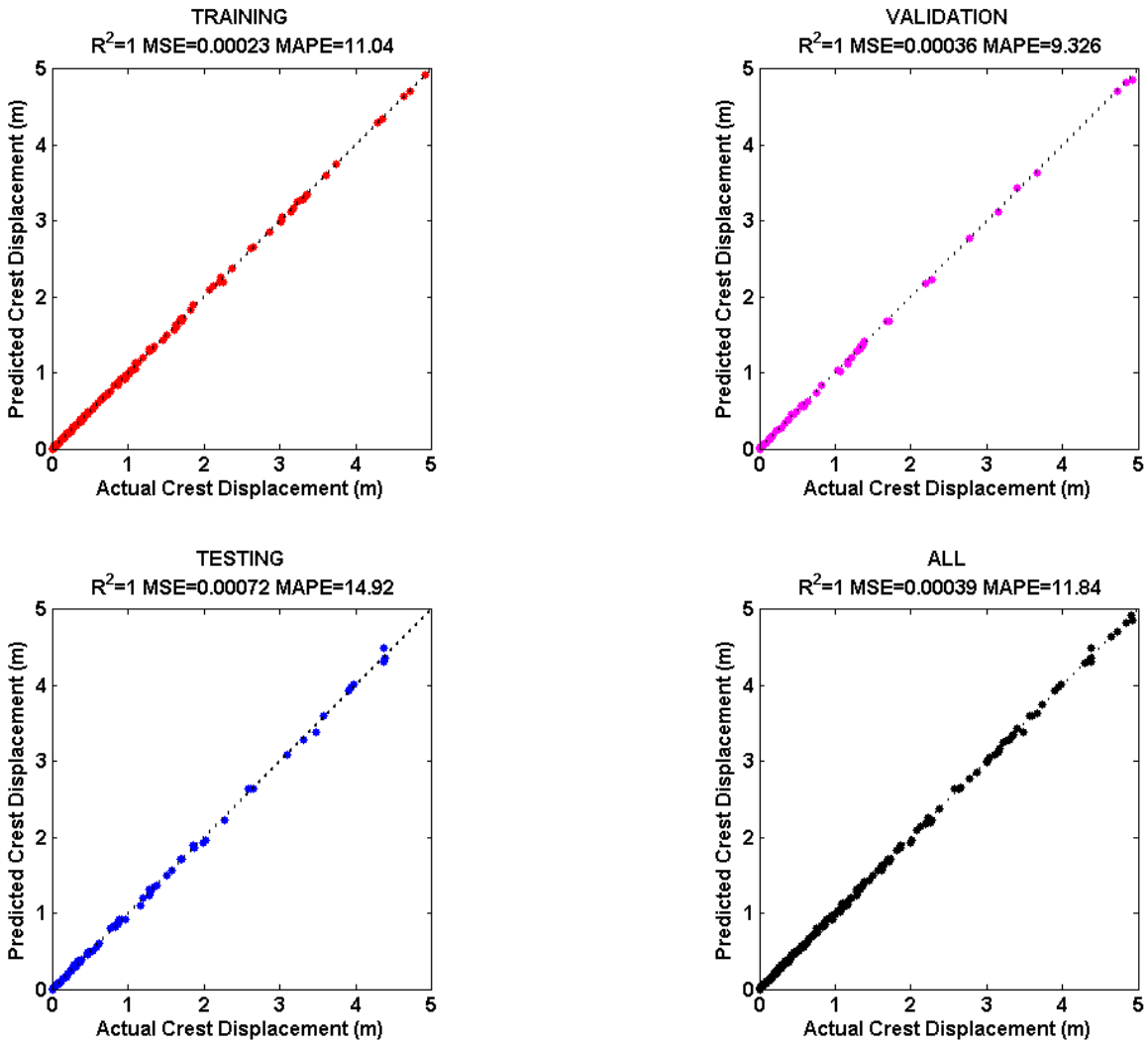


Figure 4.2 Performance of ANN model for 250 learning points for single slope case

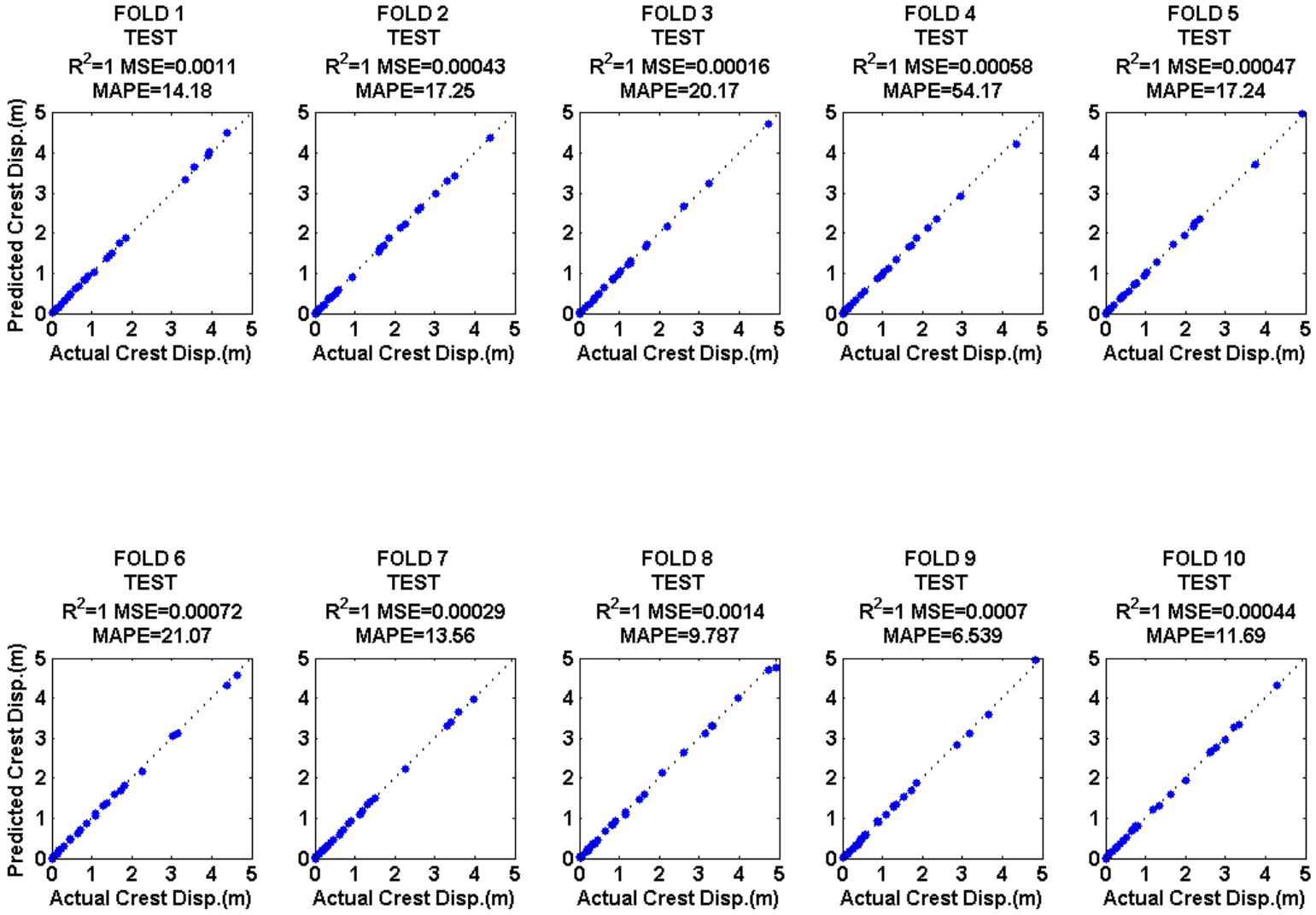


Figure 4.3 10-fold cross-validation of 350 experimental points created by artificial earthquake records for single slope case

It has been shown that ANN based metamodel performed very well for regression in the case of artificial earthquake records created from a stochastic model for a particular region. The fundamental component for the stochastic method is the spectrum of the ground motion and this is where the physics of the earthquake process and wave propagation are contained in the form of equations and parameters (Boore, 2003). Therefore, artificial earthquake records created from a single stochastic model for a specific region would have similar characteristics in terms of the amplitude of the motion, the frequency content and the duration of the motion. The reason behind this similarity is the fact that all the records have the same source and path parameters defined for the region of interest.

In order to investigate the performance of the ANN based metamodel with real earthquake records, 289 earthquake records are obtained from 40 different earthquakes. The data set includes motions from earthquakes which moment magnitudes $\boldsymbol{M_w}$ ranging from 5 to 7.68 and the distance from 0.1 to 100 km. Orthogonal components from the same stations are used as separate records. The earthquake records are obtained from Next Generation Attenuation (NGA) strong motion database of the Pacific Earthquake Engineering Center (PEER) (Power et al., 2008).

FLAC runs are performed on the same slope using 289 real earthquake records. ANN based metamodel created using artificial earthquake records is used to predict the crest displacements calculated using real earthquake records. Figure 4.4 shows the results of the ANN metamodel which uses artificial records for training/validation and real earthquake records for testing. As it can be seen from Figure 4.4, the performance is not as impressive as the case where only artificial records are used (MSE = 0.18). The reason behind this lack of performance can be explained by investigating the histograms of both real and artificial earthquake intensity measures shown in Figure 4.5. All of the intensity measures other than mean period (T_m) cover similar ranges for both artificial and real records. However, a noticeable difference can be observed for the mean periods. The

mean period range for the artificial records is from 0.25 to 0.5 seconds. On the other hand, the range of mean periods for the real earthquake records is from 0.1 to 1.3 seconds. This observation shows that the frequency content of the artificial records are in a narrow window because they are created using single stochastic model for a specific region. The ANN metamodel is trained and validated by artificial records for a narrow range of frequency content. Therefore, the metamodel is having trouble to predict the crest displacements for real earthquakes which have different frequency characteristics.

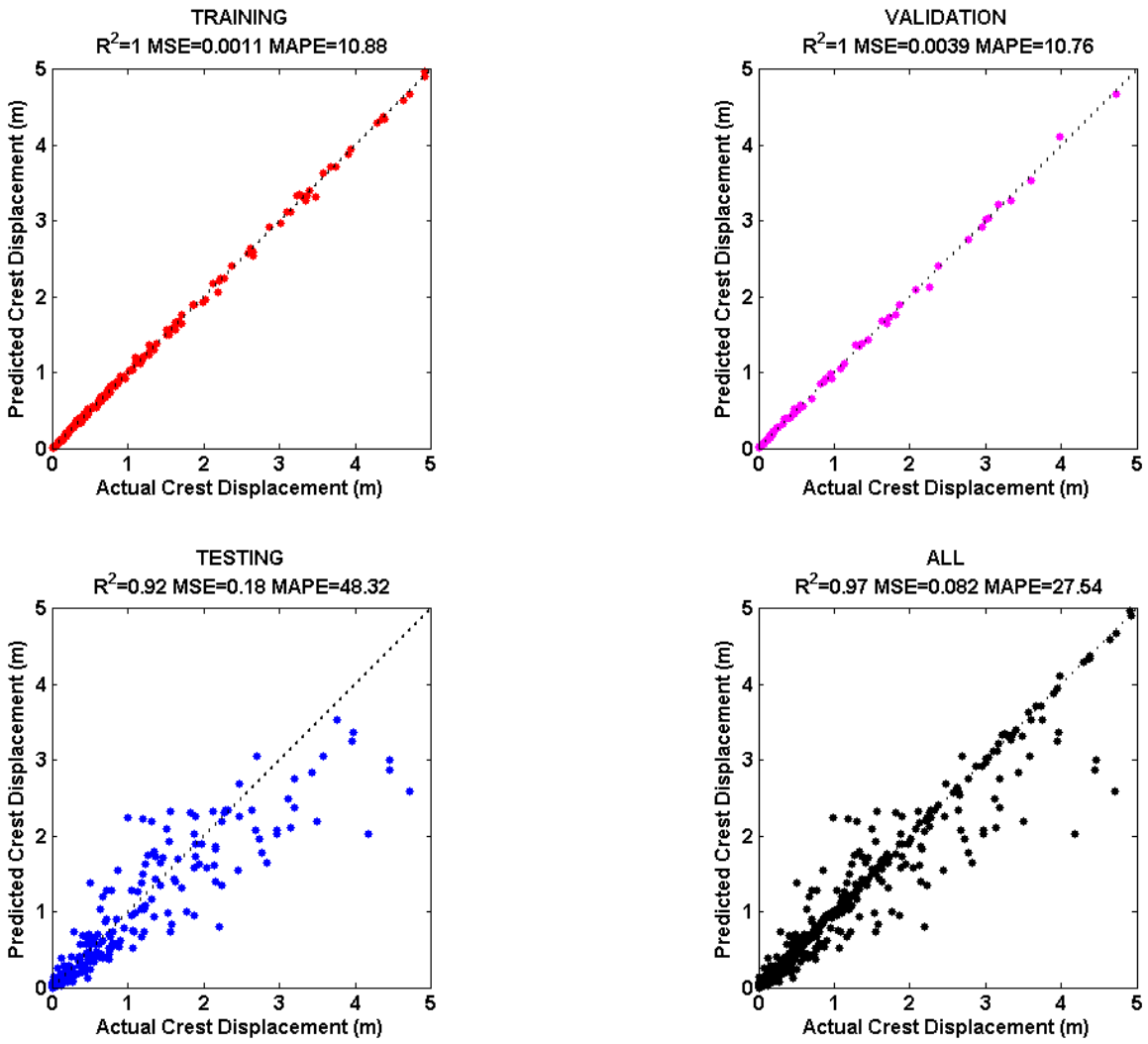


Figure 4.4 Performance of ANN model trained and validated by artificial records and tested on real records for single slope case

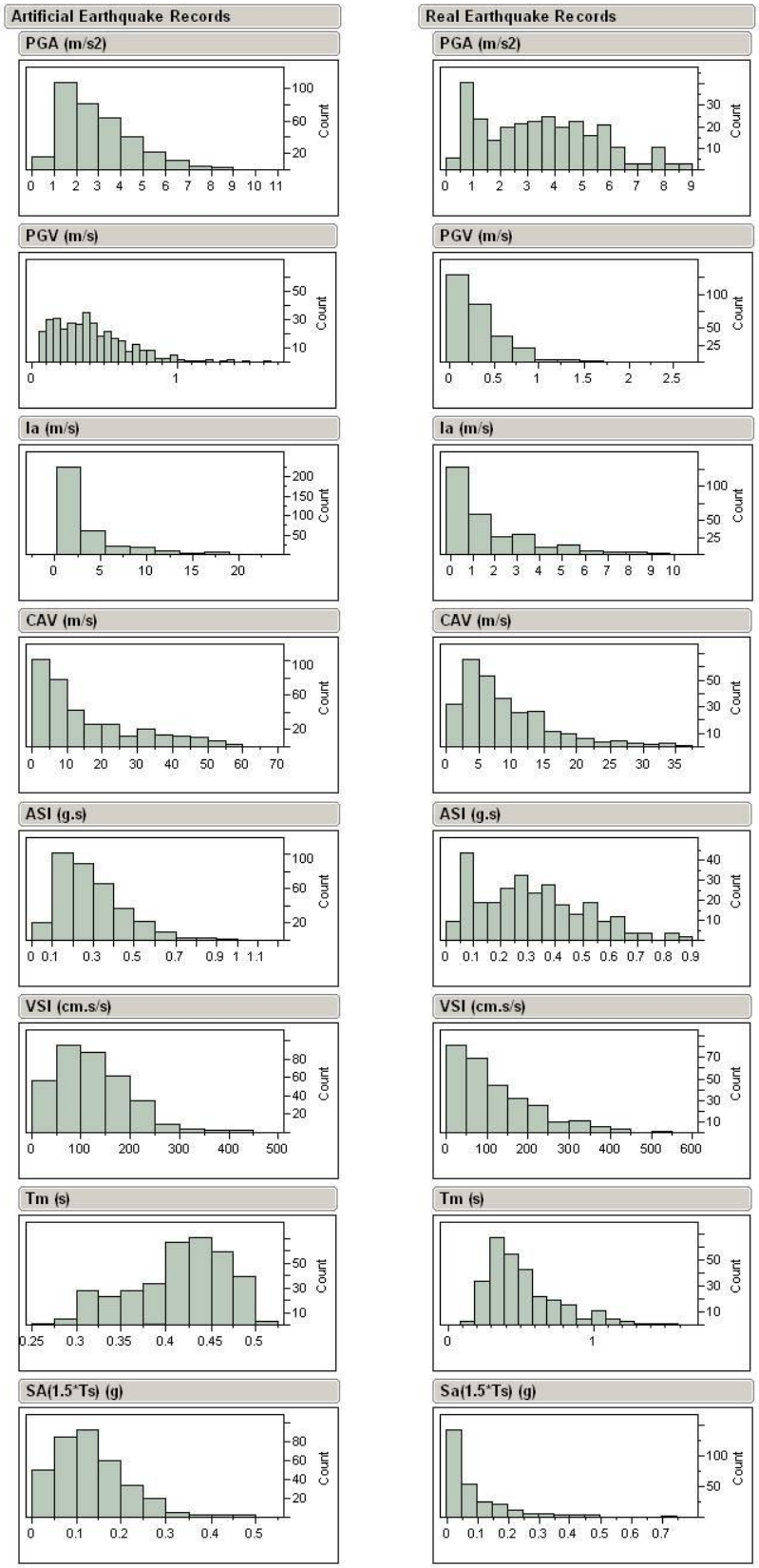


Figure 4.5 Histograms of intensity measures for artificial and real earthquake records

In order to investigate the effect of the frequency range of the input motions, an ANN model is created using the real earthquake records for training and validation. The model is than tested on the data set created using artificial earthquake records. The results can be seen in Figure 4.6. The performance of the model on predicting the test set improved considerably (MSE = 0.013). This improvement is a result of training/validating the model with a wider range of frequency content and testing it on a narrow range of frequency content which is also a subset of the wider frequency range.

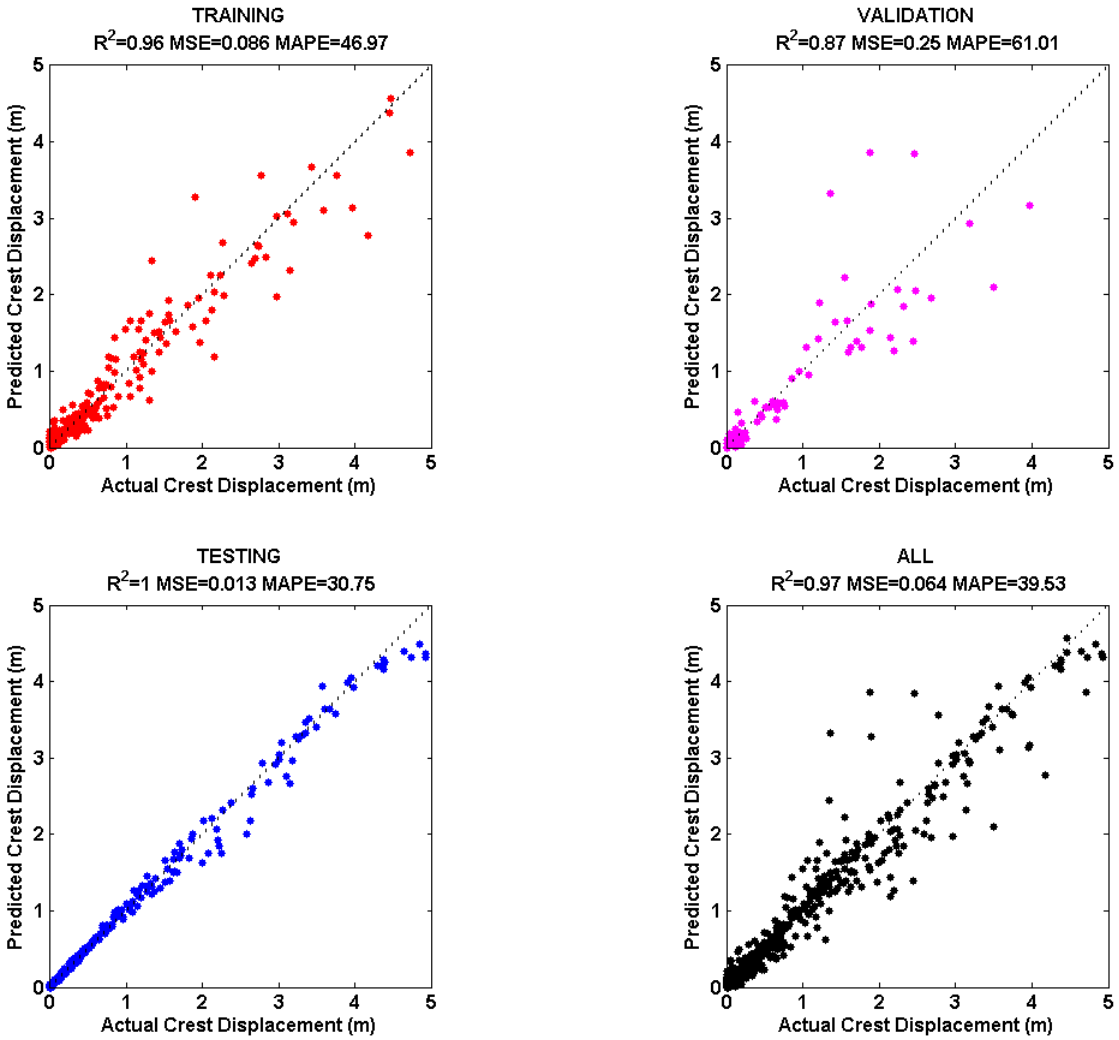


Figure 4.6 Performance of ANN model trained and validated by real records and tested on artificial records for single slope case

The calculated crest displacements from FLAC analysis can be viewed appropriately as an index of the seismic performance of the slope. However, these calculations will always be approximations because of the complexity of the dynamic response of the slope materials and the variability of the strong ground motion (Bray and Travasarou, 2007). Therefore, in order to simplify the problem at hand, crest displacements can be classified into different hazard classes and the problem can be treated as a classification problem. In most soils, 5 cm is regarded as a critical displacement leading to macroscopic ground cracking and general failure of landslides (Jibson, 1993). For that reason, crest displacements smaller than 5 cm are assigned the hazard class 'Low'. Crest displacements between 5 cm and 50 cm are assigned "Medium" and higher than 50 cm "High" (Bray and Travasarou, 2007).

The same data set which consists of artificial and real earthquake records applied to a single slope is used to investigate the performance of the metamodel to predict the hazard class. The problem is modeled as a three class classification problem. The displacement values at the crest are assigned one of the three hazard classes given in Table 4.6. Support vector machines (SVM) is used as classifier. Radial basis function (RBF) is used as a kernel. Grid search method explained in section 3.5.3.3 is used to determine the optimum values of C and γ that give the best prediction accuracy. These parameters are calibrated using cross-validation on the learning data. Learning and testing point inputs are scaled so that they fall in the range [0, 1].

Table 4.6 Assigned hazard classes for crest displacements

Crest Displacement ≤ 5 cm	Low
5 cm < Crest Displacement ≤ 50 cm	Medium
50 cm < Crest Displacement	High

First, the SVM classifier is trained using the artificial earthquake records and tested on the data set created with real earthquake records. Table 4.7 shows the confusion matrices for both training and testing sets. 80.28% of the hazard levels calculated using real earthquake records are accurately predicted using the SVM classifier trained by the hazard levels calculated by artificial earthquake records. Also it can be seen in the confusion matrix for test data that none of the high hazard levels is predicted as low hazard level and none of the low hazard levels is predicted as high hazard level.

Table 4.7 Confusion matrices of SVM model trained and validated by artificial records and tested on real records (using all 8 IM) for single slope case

Training

ACTUAL \ PREDICTED	L	M	H
L	71	0	0
M	0	121	0
H	0	1	157
	99.71%		

Testing

ACTUAL \ PREDICTED	L	M	H
L	50	6	0
M	27	71	5
H	0	19	111
	80.28%		

Furthermore, a feature selection process is carried out to determine the best possible combination of intensity measures to be used as inputs to the classifier which gives the highest prediction accuracy on the test samples. Feature selection reduces the dimensionality of data by selecting only a subset of input variables to create a model. Since, the total number of intensity measures used as inputs is relatively low (8), an exhaustive search can be performed for feature selection (Jain et al., 2000). Exhaustive search evaluates all possible subsets (combinations) of features that could be used as inputs to the classifier and chooses the best combination for each total number of features that gives the best prediction accuracy. Table 4.8 shows the results of the exhaustive

search performed on the SVM classifier trained by artificial records and tested on real earthquake records. For each number of features up to a total of 8 features, all possible combinations are tested and the intensity measures that give the best prediction accuracy are shown in the table. For example, using only one intensity measure out of eight intensity measures, VSI gives the highest prediction accuracy (75.09%). From all of the combinations the best prediction accuracy (85.81%) is obtained using Ia, CAV, VSI and Sa(1.5*Ts) as inputs. This is a considerable increase (5.5%) from using all intensity measures. The confusion matrices for the SVM classifier using only Ia, CAV, VSI and Sa(1.5*Ts) as inputs can be seen in Table 4.9.

Table 4.8 Exhaustive search for the input features to the SVM classifier trained with artificial records and tested on real records for single slope case

Number of Features	Prediction Accuracy (%)	Intensity Measures (IM)
1	75.09	VSI
2	80.62	Ia, T_m
3	85.12	PGA, CAV, VSI
4	85.81	Ia, CAV, VSI, Sa(1.5*Ts)
5	84.43	PGA, PGV, Ia, CAV, T_m
6	84.08	PGA, PGV, Ia, CAV, ASI, T_m
7	82.70	PGA, PGV, Ia, CAV, ASI, VSI, T_m
8	80.28	PGA, PGV, Ia, CAV, ASI, VSI, T_m, Sa(1.5*Ts)

Subsequently, in a similar manner to the regression metamodel, the SVM classifier is trained using the real earthquake records and then tested on the data set created with artificial earthquake records. Table 4.10 shows the confusion matrices for both training and testing sets. Comparable to the results of regression metamodels, the SVM classifier trained by real earthquake records and tested on artificial records

performed noticeably better (93.43%) than the classifier trained by the artificial records and tested on real earthquake records.

Table 4.9 Confusion matrices of SVM model trained and validated by artificial records and tested on real records (using Ia, CAV, VSI, Sa(1.5*Ts)) for single slope case

PREDICTED

ACTUAL	L	M	H
L	71	0	0
M	0	121	0
H	0	0	158
	100%		

Training

PREDICTED

ACTUAL	L	M	H
L	54	2	0
M	12	88	3
H	0	24	106
	85.81%		

Testing

Table 4.10 Confusion matrices of SVM model trained and validated by real records and tested on artificial records (using all 8 IM) for single slope case

PREDICTED

ACTUAL	L	M	H
L	54	2	0
M	7	91	5
H	0	5	125
	93.43%		

Training

PREDICTED

ACTUAL	L	M	H
L	66	5	0
M	0	103	18
H	0	0	158
	93.43%		

Testing

An exhaustive search is performed for the SVM classifier trained by real earthquake records and tested on artificial earthquake records. Table 4.11 shows the results of the exhaustive search. SVM metamodel trained with CAV as a single input parameter gives 85.71% prediction accuracy. The highest prediction accuracy (96.57%) is obtained using all intensity measures except PGA. The second best prediction accuracy

is obtained at 95.43% by only using three intensity measures PGA, CAV, T_m. The confusion matrices of the SVM classifier that gives the highest prediction accuracy can be seen in Table 4.12.

Table 4.11 Exhaustive search for the input features to the SVM classifier trained with real records and tested on artificial records for single slope case

Number of Features	Prediction Accuracy (%)	Intensity Measures (IM)
1	85.71	CAV
2	90.29	Ia, CAV
3	95.43	PGA, CAV, T_m
4	92.00	PGA, PGV, CAV, T_m
5	90.86	PGA, PGV, CAV, ASI, T_m
6	93.43	PGV, Ia, CAV, ASI, T_m, Sa(1.5*Ts)
7	96.57	PGV, Ia, CAV, ASI, VSI, T_m, Sa(1.5*Ts)
8	93.43	PGA, PGV, Ia, CAV, ASI, VSI, T_m, Sa(1.5*Ts)

Table 4.12 Confusion matrices of SVM model trained and validated by real records and tested on artificial records (using PGV, Ia, CAV, ASI, VSI, T_m, Sa(1.5*Ts)) for single slope case

PREDICTED

ACTUAL	L	M	H
L	55	1	0
M	9	88	6
H	0	6	124
	92.39%		

Training

PREDICTED

ACTUAL	L	M	H
L	65	6	0
M	0	116	5
H	0	1	157
	96.57%		

Testing

4.4 Predicting Seismic Responses of Different Slopes under Different Earthquake Loads

In this section, the performances of homogenous dry slopes with different geometric and material properties are investigated using both real and artificial earthquake records. A metamodeling approach is adopted to approximate the seismic performances of the slopes as a function of geometric and material properties and earthquake loading. Six parameters (design variables) are chosen for the geometric and material properties of the slopes. The parameters selected, include the inclination of slope ($\boldsymbol{\beta}$), the height of the slope ($\boldsymbol{H}$), the effective friction angle of soil($\boldsymbol{\phi'}$), the density of soil ($\boldsymbol{d}$), the effective cohesion of soil ($\boldsymbol{c'}$) and the shear wave velocity of the soil ($\boldsymbol{Vs}$). Earthquake magnitude **(Mw)** and distance from the epicenter of the earthquake **(R)** are chosen as strong motion design variables. The artificial earthquake records are created using SMSIM for given magnitude and distance combinations. Similar to the artificial records created for the case of a single slope, the Atkinson and Silva (2000) stochastic model for California is used for the SMSIM input parameters. The source parameters for the Atkinson and Silva (2000) model can be seen in Table 4.3. The parameter ranges to be used in design of experiments are shown in Table 4.13.

A sequential Latin hypercube sampling is used for the design of experiments. "R" statistical software's "Latin Hypercube Samples" package is utilized to create the experimental points using a *maximin* Latin hypercube design. Initially, a run size of 100 experiments is created. However, unlike the experimental designs shown in Chapter 3 and earlier in this chapter, not all of the created experiments can be used for creating the metamodels, because some of the experiments designed will have factor of safety (F.S.) values smaller than 1. Namely, these experiments will be statically unstable and a seismic analysis cannot be carried out with an unstable slope. For this reason, after creating the initial experiments, the slopes with F.S. values smaller than 1 should be removed from

the data set before continuing with the seismic analysis. The metamodels created in Chapter 3 are shown to predict the F.S. values with reasonable accuracy. Therefore, instead of calculating the F.S. values for the designed experiments which would be computationally expensive, the ANN regression metamodel created in Chapter 3 is used to predict the F.S. values. As it can be seen in Table 3.1 and Table 4.13 assigned ranges for the design variables are the same for both cases with the exception of the shear wave velocity of the soil ($\boldsymbol{Vs}$) which does not affect the factor of safety of the slope. From the 100 experiments initially created 52 have predicted F.S. values higher than 1. Subsequently, more experiments are augmented to the original space in increments of 100 experimental points. In most of the increments, about 50% of the designed experiments had F.S. values higher than 1.

Table 4.13 Assigned ranges for design variable values

	Design Variables							
	$\boldsymbol{H}$	$\boldsymbol{\beta}$	$\boldsymbol{\phi'}$	$\boldsymbol{c'}$	$\boldsymbol{d}$	**Vs**	**Mw**	**R**
Lower bound	10 m	10^o	20^o	5 kPa	1600 kg/m^3	300 m/sec	6	5 km
Upper bound	350 m	60^o	50^o	50 kPa	2500 kg/m^3	800 m/sec	8.5	30 km

FLAC simulations are performed for each designed experiment (with predicted F.S.>1) using the geometric and material properties assigned and earthquake records created using SMSIM for different Mw and R combinations.

The displacement value calculated at the crest of the slope using FLAC is assigned as the damage index of the slope. The problem is modeled as a classification problem. The displacement value at the crest is assigned one of the three hazard classes given in Table 4.6. Support vector machines (SVM) is used as classifier. SVM classifier is trained using 15 input parameters. 15 input parameters consist of 6 geometric and

material parameters given in Table 4.13, 8 intensity measures for characterization of the earthquake records given in Table 4.4 and the factor of safety predicted by the metamodel created in Chapter 3.

The first 100 experiments that have F.S. values higher than 1 are assigned as the test set and are not included in the training of the models and they are only used for testing the predictive accuracy of the metamodels. The SVM classifier is trained using the first batch of 50 experimental points (F.S.>1) and the performance is evaluated using 100 test points. The performance of the SVM classifier for the first 50 training points using can be seen in Table 4.14. The prediction accuracy is calculated to be 72% when the SVM classifier is trained only with 50 training points.

Table 4.14 Confusion matrices of SVM model trained by 50 training points

PREDICTED

ACTUAL	L	M	H
L	17	0	0
M	1	16	0
H	0	0	16
	98%		

Training

PREDICTED

ACTUAL	L	M	H
L	23	15	0
M	2	27	2
H	0	9	22
	72%		

Testing

Next, the performance of the SVM classifier is evaluated while increasing the number of training points. Table 4.15 shows the performance of the SVM classifier with increasing training points. Prediction accuracy of the testing samples starts with 72% and increase up to 85% with increasing training points. The prediction accuracy reaches a maximum at 350 training points and does not improve further. Also the prediction accuracy of the training samples starts with a very high accuracy of 98 percent and gradually drops to about 85%. Because for low number of training points SVM classifier

is prone to overfitting and as the number of training points increase the accuracy becomes less biased. All of the confusion matrices of SVM classifiers for increasing numbers of training points can be seen in Appendix B (Table B.1-B.7).

Table 4.15 Performance summary of SVM Classifier with increasing number of training points

Number of Training Points	Training	Testing
	Prediction Accuracy (%)	Prediction Accuracy (%)
50	98	72
100	99	76
150	87.3	79
200	85.5	81
250	81.2	82
300	85.3	83
350	83.7	85
400	86	84
450	85.6	84

The confusion matrices of the SVM classifier for 350 training points using can be seen in Table 4.16. The prediction accuracy is calculated to be 85%.

Table 4.16 Confusion matrices of SVM model trained by 350 training points

PREDICTED

ACTUAL	L	M	H
L	115	13	0
M	18	88	6
H	4	16	90
	83.71%		

Training

PREDICTED

ACTUAL	L	M	H
L	33	5	0
M	3	25	3
H	0	4	27
	85%		

Testing

A feature selection procedure is carried out to determine if it is possible to improve the prediction accuracy of the SVM classifier by using a subset of 15 input parameters. The exhaustive search method introduced earlier in the case of single slopes is guaranteed to find the optimal subset of input parameters that gives the best prediction accuracy. However, exhaustive search is not feasible for higher number of input parameters because of the extremely high number of possible feature combinations. For example for the case of the single slope, there are total of 8 input parameters which give $\binom{8}{1} + \binom{8}{2} \cdots + \binom{8}{8} = 255$ possible feature combinations and it is computationally feasible to train and test the SVM classifier for each combination. However, with 15 input parameters there are $\binom{15}{1} + \binom{15}{2} \cdots + \binom{15}{15} = 32767$ possible combinations. Therefore, it is not feasible to perform an exhaustive search. However, there are other alternative feature selection methods available.

Sequential Forward Selection (SFS) is a common method of feature selection for the cases where the number of predictor variables is high (Jain et al., 2000). In SFS, features are sequentially added to an empty candidate set. The algorithm tries all single features and selects the feature that gives the best prediction accuracy and retains that feature. Next, the algorithm tries adding each one of the remaining features to the candidate set and again chooses the one that gives the best prediction accuracy together with the first feature retained. The process continues until all the features are added to the set. Then, the step which has the highest prediction accuracy is chosen.

Sequential Backward Selection (SBS) is also a common method of feature selection (Jain et al., 2000). In SBS, the algorithm starts with all input features included in the candidate set and successively removes one feature at a time. The feature removed is selected so that the accuracy of the classifier is higher with the removal of that feature than any other feature. The method continues until all but one feature are removed. The set which has the highest prediction accuracy is chosen.

Since both Sequential Forward Selection (SFS) and Sequential Backward Selection (SBS) are computationally cost effective, they can both be utilized at the same time and the one which gives better results can be chosen. However, it is important to note that these methods do not guarantee the optimal solution because when a feature is retained in SFS it cannot be discarded and when a feature is deleted in SBS it cannot be brought back. Table 4.17 shows the SFS results for the SVM classifier trained by 350 training points. As it can be seen from Table 4.17 the best single input parameter is PGA which gives 72% prediction accuracy. The maximum prediction accuracy is obtained with 9 features at 88%.

Table 4.17 SFS for the input features to the SVM classifier trained by 350 training points

Number of Features	Prediction Accuracy (%)	Input Parameters
1	72	PGA
2	74	β, PGA
3	83	β, F.S., PGA
4	84	β, F.S., PGA, ASI
5	85	β, F.S., PGA, ASI, T_m
6	85	β, F.S., PGA, Ia, ASI, T_m
7	85	β, φ, F.S., PGA, Ia, ASI, T_m
8	86	β, φ, d, F.S., PGA, Ia, ASI, T_m
9	88	β, H, φ, d, F.S., PGA, Ia, ASI, T_m
10	86	β, H, φ, c, d, F.S., PGA, Ia, ASI, T_m
11	85	β, H, φ, c, d, V_s, F.S., PGA, Ia, ASI, T_m
12	85	β, H, φ, c, d, V_s, F.S., PGA, Ia, ASI, VSI, T_m
13	87	β, H, φ, c, d, V_s, F.S., PGA, Ia, ASI, VSI, T_m, Sa(1.5*Ts)
14	85	β, H, φ, c, d, V_s, F.S, PGA, PGV, Ia, ASI, VSI, T_m, Sa(1.5*Ts)
15	85	β, H, φ, c, d, V_s, F.S, PGA, PGV, Ia, CAV, ASI, VSI, T_m, Sa(1.5*Ts)

Table 4.18 shows the SBS results for the SVM classifier trained by 350 training points. As it can be seen from Table 4.18, like SFS the best single input parameter is found to be PGA which gives 72% prediction accuracy. The maximum prediction accuracy is obtained with 7 features at 89%. SBS performed slightly better than SFS. It is also important to note that both SFS and SBS yielded good prediction accuracies even with relatively small number of input parameters. For example, SBS results show that with only using 4 input parameters (β, φ, PGA, T_m) 88% prediction accuracy is obtained.

Table 4.18 SBS for the input features to the SVM classifier trained by 350 training points

Number of Features	Prediction Accuracy (%)	Input Parameters
15	85	β, H, φ, c, d, V_s, F.S, PGA, PGV, Ia, CAV, ASI, VSI, T_m, Sa(1.5*Ts)
14	85	β, H, φ, c, d, V_s, F.S, PGA, PGV, CAV, ASI, VSI, T_m, Sa(1.5*Ts)
13	86	β, H, φ, c, d, V_s, F.S, PGA, PGV, CAV, ASI, T_m, Sa(1.5*Ts)
12	85	β, H, φ, c, d, V_s, PGA, PGV, CAV, ASI, T_m, Sa(1.5*Ts)
11	87	β, H, φ, c, d, V_s, PGA, PGV, CAV, ASI, T_m
10	86	β, H, φ, c, d, PGA, PGV, CAV, ASI, T_m
9	86	β, φ, c, d, PGA, PGV, CAV, ASI, T_m
8	87	β, φ, c, d, PGA, PGV, ASI, T_m
7	89	β, φ, d, PGA, PGV, ASI, T_m
6	87	β, φ, PGA, PGV, ASI, T_m
5	87	β, φ, PGA, ASI, T_m
4	88	β, φ, PGA, T_m
3	78	β, φ, PGA
2	74	β, PGA
1	72	PGA

Next, similar to the single slope case, the performance of the SVM based metamodel trained with artificial earthquake records are tested on a data set of numerical experiment results created by real earthquake records. The same 289 earthquake records

used in the single slope case are employed. A number of experimental points are created using a *maximin* Latin hypercube design. The number is chosen so that at least 289 of the experiments would have factor of safety values higher than 1 and therefore can be used in seismic analysis. Only 6 design variables which include the geometric and material parameters of the slope are used for the design of experiments. Mw and R are excluded because no artificial earthquake records are created. 289 real earthquake records are randomly assigned to each of the designed experiments with F.S.>1. FLAC analyses are performed for each of the experiment and the hazard values are assigned to each slope according to the crest displacements calculated.

SVM classifier based metamodel is built and trained by all 550 experiments (100 test set + 450 training set) created by artificial earthquake records and tested on the 289 experimental points generated by real earthquake records. Table 4.19 shows the confusion matrices for the metamodel. The prediction accuracy is close to 80%.

Table 4.19 Confusion matrices of SVM model trained by artificial earthquake records and tested on real earthquake records

		PREDICTED		
		L	M	H
ACTUAL	L	181	20	0
	M	28	135	12
	H	1	30	143
		83.46%		

Training

		PREDICTED		
		L	M	H
ACTUAL	L	91	22	0
	M	11	78	14
	H	0	12	61
		79.59%		

Testing

In order to improve the prediction accuracy of the metamodel both SFS and SBS are performed on the SVM classifier. The results of the SFS and SBS can be seen in Tables 4.20 and 4.21. The best prediction accuracy is found using SBS at 84.43%. SBS provided almost 5% increase in the prediction accuracy. SFS and SBS results also show that it is possible to obtain prediction accuracies over 80% using only a relatively small number of input parameters. For example, as it can be seen in the SBS results 83.39% prediction accuracy can be obtained with just using 4 input parameters (φ, F.S, VSI, Sa(1.5*Ts)). The confusion matrices for the SVM metamodel which gives the best prediction accuracy of 84.43% can be seen in Table 4.22.

Table 4.20 SFS for the input features to the SVM classifier trained by artificial earthquake records and tested on real earthquake records

Number of Features	Prediction Accuracy (%)	Input Parameters
1	63.67	VSI
2	73.01	F.S, VSI
3	81.66	F.S, VSI, Sa(1.5*Ts)
4	83.39	φ, F.S, VSI, Sa(1.5*Ts)
5	83.05	φ, F.S, Ia, VSI, Sa(1.5*Ts)
6	83.74	H, φ, F.S, Ia, VSI, Sa(1.5*Ts)
7	82.01	H, φ, F.S, PGA, Ia, VSI, Sa(1.5*Ts)
8	82.70	β, H, φ, F.S, PGA, Ia, VSI, Sa(1.5*Ts)
9	83.05	β, H, φ, V_s, F.S, PGA, Ia, VSI, Sa(1.5*Ts)
10	83.74	β, H, φ, c, V_s, F.S, PGA, Ia, VSI, Sa(1.5*Ts)
11	81.32	β, H, φ, c, V_s, F.S, PGA, Ia, CAV, VSI, Sa(1.5*Ts)
12	83.39	β, H, φ, c, V_s, F.S, PGA, Ia, CAV, ASI, VSI, Sa(1.5*Ts)
13	79.93	β, H, φ, c, d, V_s, F.S, PGA, Ia, CAV, ASI, VSI, Sa(1.5*Ts)
14	78.20	β, H, φ, c, d, V_s, F.S, PGA, PGV, Ia, CAV, ASI, VSI, Sa(1.5*Ts)
15	79.59	β, H, φ, c, d, V_s, F.S, PGA, PGV, Ia, CAV, ASI, VSI, T_m, Sa(1.5*Ts)

Table 4.21 SBS for the input features to the SVM classifier trained by artificial earthquake records and tested on real earthquake records

Number of Features	Prediction Accuracy (%)	Input Parameters
15	79.59	β, H, φ, c, d, V_s, F.S, PGA, PGV, Ia, CAV, ASI, VSI, T_m, Sa(1.5*Ts)
14	80.62	β, H, φ, c, V_s, F.S, PGA, PGV, Ia, CAV, ASI, VSI, T_m, Sa(1.5*Ts)
13	80.28	β, H, φ, c, F.S, PGA, PGV, Ia, CAV, ASI, VSI, T_m, Sa(1.5*Ts)
12	81.32	β, H, φ, c, F.S, PGA, PGV, Ia, CAV, ASI, VSI, Sa(1.5*Ts)
11	83.39	β, H, φ, c, F.S, PGA, Ia, CAV, ASI, VSI, Sa(1.5*Ts)
10	84.43	β, H, φ, c, F.S, Ia, CAV, ASI, VSI, Sa(1.5*Ts)
9	83.74	β, φ, c, F.S, Ia, CAV, ASI, VSI, Sa(1.5*Ts)
8	84.08	β, φ, c, F.S, Ia, CAV, VSI, Sa(1.5*Ts)
7	84.08	β, φ, c, F.S, CAV, VSI, Sa(1.5*Ts)
6	84.08	β, φ, c, F.S, VSI, Sa(1.5*Ts)
5	82.70	β, φ, F.S, VSI, Sa(1.5*Ts)
4	83.39	φ, F.S, VSI, Sa(1.5*Ts)
3	81.66	F.S, VSI, Sa(1.5*Ts)
2	74.05	F.S, Sa(1.5*Ts)
1	61.25	Sa(1.5*Ts)

Table 4.22 Confusion matrices of SVM model trained by artificial earthquake records and tested on real earthquake records using input parameters β, H, φ, c, F.S, Ia, CAV, ASI, VSI, Sa(1.5*Ts)

PREDICTED

ACTUAL	L	M	H
L	185	16	0
M	19	148	8
H	0	12	162
	90.00%		

Training

PREDICTED

ACTUAL	L	M	H
L	103	10	0
M	18	81	4
H	0	13	60
	84.43%		

Testing

Finally, in order to get a less biased estimate of the performance of the metamodel for predicting seismic responses of different slopes under different earthquake loads, a 10-fold cross-validation is performed using the combined data set involving the experimental points created using artificial earthquake records and real earthquake records. The data set consists of 550 experimental points created using artificial earthquake records (450 training set + 100 test set) and 289 real earthquake records. Using all 15 input parameters the average prediction accuracy of 10-fold cross validation is found to be 84.62%.

Moreover, sequential forward selection (SFS) and sequential backward selection are performed in order to improve the average 10-fold cross-validation prediction accuracy. SFS and SBS results can be seen in Table 4.23 and 4.24 respectively.

Table 4.23 SFS for the input features to the SVM classifier for 10-fold cross-validation using both artificial and real earthquake records

Number of Features	Prediction Accuracy (%)	Input Parameters
1	64.37	VSI
2	76.87	F.S, VSI
3	82.00	F.S, VSI, Sa(1.5*Ts)
4	83.07	H, F.S, VSI, Sa(1.5*Ts)
5	84.02	β, H, F.S, VSI, Sa(1.5*Ts)
6	84.86	β, H, F.S, CAV, VSI, Sa(1.5*Ts)
7	85.45	β, H, F.S, CAV, VSI, T_m, Sa(1.5*Ts)
8	85.45	β, H, F.S, Ia, CAV, VSI, T_m, Sa(1.5*Ts)
9	85.21	β, H, F.S, PGV, Ia, CAV, VSI, T_m, Sa(1.5*Ts)
10	85.09	β, H, V_s, F.S, PGV, Ia, CAV, VSI, T_m, Sa(1.5*Ts)
11	84.98	β, H, V_s, F.S, PGA, PGV, Ia, CAV, VSI, T_m, Sa(1.5*Ts)
12	84.62	β, H, V_s, F.S, PGA, PGV, Ia, CAV, ASI, VSI, T_m, Sa(1.5*Ts)
13	84.38	β, H, d, V_s, F.S, PGA, PGV, Ia, CAV, ASI, VSI, T_m, Sa(1.5*Ts)
14	83.91	β, H, c, d, V_s, F.S, PGA, PGV, Ia, CAV, ASI, VSI, T_m, Sa(1.5*Ts)
15	84.62	β, H, φ, c, d, V_s, F.S, PGA, PGV, Ia, CAV, ASI, VSI, T_m, Sa(1.5*Ts)

The highest 10-fold cross-validation accuracy (86.83%) is obtained by SBS with 11 input features. 82% accuracy is obtained by just using three parameters (F.S, VSI, Sa(1.5*Ts)).

Table 4.24 SBS for the input features to the SVM classifier for 10-fold cross-validation using both artificial and real earthquake records

Number of Features	Cross-validation Accuracy (%)	Input Parameters
15	84.62	β, H, φ, c, d, V_s, F.S, PGA, PGV, Ia, CAV, ASI, VSI, T_m, Sa(1.5*Ts)
14	85.10	β, H, φ, c, d, V_s, F.S, PGA, PGV, Ia, CAV, ASI, VSI, Sa(1.5*Ts)
13	85.44	β, H, φ, c, d, V_s, F.S, PGV, Ia, CAV, ASI, VSI, Sa(1.5*Ts)
12	86.75	β, H, φ, c, d, V_s, F.S, PGV, CAV, ASI, VSI, Sa(1.5*Ts)
11	86.83	β, H, φ, c, V_s, F.S, PGV, CAV, ASI, VSI, Sa(1.5*Ts)
10	85.46	β, φ, c, V_s, F.S, PGV, CAV, ASI, VSI, Sa(1.5*Ts)
9	84.98	β, φ, c, V_s, F.S, PGV, CAV, VSI, Sa(1.5*Ts)
8	84.86	β, φ, c, V_s, F.S, CAV, VSI, Sa(1.5*Ts)
7	83.79	β, φ, c, V_s, F.S, VSI, Sa(1.5*Ts)
6	83.43	φ, c, V_s, F.S, VSI, Sa(1.5*Ts)
5	83.11	φ, V_s, F.S, VSI, Sa(1.5*Ts)
4	82.84	V_s, F.S, VSI, Sa(1.5*Ts)
3	82.00	F.S, VSI, Sa(1.5*Ts)
2	76.87	F.S, VSI
1	64.37	VSI

4.4 Summary

The soft computing methodology introduced in Chapter 3 is applied to case of dynamic response analysis of slopes. A finite difference method (FLAC) is used to calculate the permanent deformation of slopes under earthquake loading.

First, the performance of a homogenous dry slope is investigated under different earthquake loads which consist of both synthetic and real earthquake records. While keeping all the other parameters constant, different earthquake loads are applied to the same slope and crest displacements are calculated for each earthquake load. A metamodeling approach is adopted. Earthquake magnitude **(Mw)** and distance from the epicenter of the earthquake **(R)** are chosen as design variables and artificial earthquake records are generated with SIMSIM using a stochastic model for California. A number of intensity measures (IM) are selected to characterize the earthquake time history. These intensity measures are used as inputs to the metamodels. First the problem is modeled as a regression problem where the objective is to predict the crest displacement for a given intensity measures combination. A sequential Latin hypercube sampling is used for the design of experiments. Metamodels are created using ANN regressors. ANN regressors performed very well for predicting the crest displacements using only artificial records. The average 10-fold cross validation MSE is calculated as 0.00064. Next, the performance of the metamodel created using artificial earthquakes is investigated by computer experiments conducted using real earthquake records. Mean square error (MSE) is calculated as 0.18 for the prediction of experiments performed using real earthquake records. This lack of performance is explained by the different range of mean periods for artificial and real earthquake records.

Subsequently, the problem is modeled as a classification problem where the crest displacements are classified into different hazard classes. Metamodels are created using SVM classifiers. First, the SVM classifier is trained using the artificial earthquake records and tested on the data set created with real earthquake records. Using an

exhaustive feature selection method, the best combination of input features (IMs) that gives the best prediction accuracy is determined. The highest prediction accuracy is calculated to be 85.81%. Next, the SVM classifier is trained using the real earthquake records and then tested on the data set created with artificial earthquake records. A prediction accuracy of 96.57% is obtained.

Finally, the performances of homogenous dry slopes with different geometric and material properties are investigated using both real and artificial earthquake records. Addition to **(Mw)** and **(R)** used as inputs for creating artificial earthquake records, six parameters (design variables) are chosen for the geometric and material properties of the slopes. A sequential Latin hypercube sampling is used for the design of experiments. The problem is modeled as a classification problem and SVM classifiers are used. Sequential Forward Selection (SFS) and Sequential Backward Selection (SBS) are used for feature selection because of the high number of input parameters. First, only artificial records are used to investigate the performance of the metamodels 89% percent prediction accuracy is obtained. Next the SVM classifier is trained by the artificial earthquake records and tested on the real earthquake records and 84.43% prediction accuracy is calculated. At last, a 10-fold cross-validation is performed on the whole data set including artificial and real earthquake records, and 86.83% average cross-validation accuracy is calculated.

An important point, worth underlining, is that even though the prediction accuracies are in the range of 85% to 90%, misclassifications are always in between neighboring classes. Namely, a “low” hazard class is never predicted as “high” hazard class and a “high” hazard class is never predicted as “low” hazard class.

The computer run times for the dynamic FLAC analysis depend on the number of elements in the model as wells as the duration of the input acceleration time history. The number of elements in the model depends on the shear wave velocity of the material, the frequency characteristics of the earthquake and the size of the slope. But to give an idea, for example, for a 100 meters high slope with 11000 elements and an earthquake record

of 30 seconds, the analysis takes about 60 minutes on a 2.8 GHz Intel® Core™ i7 computer.

Using the soft computing methodology, after creating the metamodels, reliable predictions of the performances of a large number of slopes with different geometric and material properties under different earthquake loads can be calculated in a matter of seconds. This time efficiency of soft computing methods makes it possible to use this method in a regional seismic landslide hazard analysis.

CHAPTER 5

DELINEATION OF SLOPE PROFILES FROM DIGITAL ELEVATION MODELS FOR REGIONAL LANDSLIDE HAZARD ANALYSIS

5.1 Introducing the Concept of Two Dimensional Slope Profile

Assessment of regional landslide hazards involves the preliminary selection of a suitable mapping unit. A mapping unit can be defined as a segment of the land surface which encloses a similar set of ground conditions which differ from the neighboring units across definable boundaries (Hansen, 1984). Different methods have been proposed for creating mapping units for landslide hazard analysis. All methods fall into one of the five groups; grid-cells, terrain units, unique-condition grids, slope units and topographic units (Carrara and Guzzetti, 1995) and details of these groups can be found therein.

For GIS-based deterministic slope stability and earthquake-induced landslide analysis, the grid-cell approach has been commonly used in conjunction with the relatively simple infinite slope model. The grid-cell approach implies the division of the study region into cells of given size and assigning values for each factor taken into consideration such as slope angle, material properties and earthquake loads. The infinite slope model is a one-dimensional model which assumes an infinitely long planar slip surface on which a rigid slab of soil mass slides parallel to the ground surface. This method is suitable for analysis of a soil mass that overlies a sloping drainage barrier that may be bedrock or a less permeable and well compacted soil layer (Hammond et al., 1992). The infinite slope model together with Newmark's displacement analysis has been widely used to create seismic landslide susceptibility maps. Landslide hazard analyses using the infinite slope model are applicable to shallow landslides. The infinite slope model gives reliable results in the case of surficial landslides with depth-length ratios smaller than 0.1 (Xie et al., 2003). On the other hand, the infinite slope model cannot

satisfactorily analyze deep-seated, rotational failure (Luo, 2006). In reality, deep seated landslides are common and these types of landslides are a major cause of property damage and fatalities. Therefore, more accurate analyses are needed for hazard mapping of deep seated landslides. In the case of deep-seated landslides, two- or three-dimensional models are required to accurately analyze both static and dynamic performance of slopes. Two-dimensional analyses are performed along the representative soil profile sections while three-dimensional analyses are performed on the entire landslide body.

Numerous analytical methods for the evaluation of 2D or 3D seismic response of slopes have been proposed and they can be divided into three main categories; stress-deformation analysis, permanent deformation analysis (Newmark analysis) and pseudostatic analysis. These 2D and 3D methods are rarely used in GIS-based landslide hazard zonation because they are numerically expensive compared to 1D infinite slope models. However, with the soft computing methods introduced in Chapter 3 and Chapter 4, relatively accurate predictions of 2D static and seismic slope responses can be calculated for a regional hazard zonation. These methods can be used to predict the responses of a vast number of different slope profiles with different material properties, geometries and input earthquake motions in a more computationally efficient way.

In order to integrate the soft computing methods with geographical information systems (GIS) for landslide hazard analysis, there is a need for an automatic slope profile delineation method for 2D slope analysis and an automatic slope unit delineation method for 3D slope analysis. The methods for the extraction of primary topographic attributes such as slope, aspect, curvature, flow direction are offered by standard GIS platforms. However, the methods for integrated mapping of slope units and delineation of slope profiles are yet to be developed.

In recent years, some researchers have used 2D and 3D slope stability methods for regional landslide hazard analysis. Xie et al. (2003) used a column-based three-dimensional limit equilibrium method to calculate static safety factors for GIS based

landslide hazard assessment. Slope unit was used as the mapping unit and the entire study area was divided into slope units using GIS-based hydrology tools. Luo (2006) used 2D slope profiles with different limit equilibrium models and Newmark's method to analyze landslides triggered by the 2001 El Salvador earthquake. Luo (2006) selected a number of slope profiles from the profiles manually created using terrain and hydrology analysis tools.

The objective of this chapter of the thesis is to explore alternative methodologies to delineate 2D slope profiles from a given Digital Elevation Model (DEM). Two different approaches are presented for delineation of 2D slope profiles. Both approaches use GIS-based hydrology tools. Subsequently, an extension to these approaches that can be used specifically to delineate slope profiles along highways is presented.

5.2 Delineation of Two Dimensional Slope Profiles from Digital Elevation Models Using Hydrology Tools

Slope units can be obtained by dividing the terrain into hydrological regions between drainage and ridge lines (Carrara and Guzzetti, 1995). There are many techniques for automatically generating drainage and ridge line networks (Band, 1986; Jenson and Domingue, 1988). There are also several GIS-based hydrological analysis and modeling tools available on the market. In this study, Arc Hydro tool (Maidment, 2002) is employed for generating drainage and ridge lines. Arc Hydro is an ArcGIS tool to support water resources applications. It offers a consistent method for delineating watersheds from DEM. Watershed is a polygon that is used for separating neighboring drainage basins or catchments. In the hydrologic model, the watershed polygon represents the local maximum elevations or ridge lines.

In order to delineate watersheds from DEM, there are several steps that should be performed in sequential order. Operation of individual steps is not discussed here in detail. A simplified flowchart of the watershed delineation process can be seen in Figure 5.1. The first step is the development of a hydrologically-correct DEM, namely filling the artificial sinks which are artifacts of DEM construction. This DEM preprocessing ensures that DEM can be used for efficient watershed delineation and drainage network generation. The watershed delineation procedure is shown on a series of figures for a sample area for which the preprocessed DEM is shown in Figure 5.2. The DEM is also shown as a hillshade raster in Figure 5.3 for better visual interpretation of the area. There are two key steps after DEM preprocessing. The first one is the creation of the flow direction grid. Flow direction defines the movement of water between the terrain cells. The flow direction in Arc Hydro is based entirely on topography. It uses the most basic flow algorithm called D8. In this algorithm, flow direction is one of the eight neighboring cells with the steepest downslope gradient. The flow direction grid for sample area can be seen in Figure 5.4. As can be seen in Figure 5.4 each cell is assigned a geographical

direction of flow (East, West, etc.). Subsequent to the flow direction grid, a flow accumulation grid is created. A flow accumulation grid consists of grids that contain the number of upstream cells that drain thorough each cell. Flow accumulation grid for the sample area can be seen in Figure 5.5. In Figure 5.5 the cells with brighter colors have more number of upstream cells that drain through that cell. The second key step is to create a drainage network. Creating a drainage network is a subjective step; there can be a number of different drainage network created using the same flow accumulation raster. There are different thresholds that can be used to create drainage networks and subsequently watersheds. The threshold used in this study is the accumulation threshold. The drainage lines are defined as the connection of the cells that have higher values than a user specified threshold. The recommended value for the threshold is 1% of the maximum flow accumulation raster. The procedure of determining and assigning the threshold is called stream definition and stream segmentation. Figure 5.6 shows the stream segmentation for the sample area. The accumulation threshold is taken as 3000 cells for the stream segmentation shown in Figure 5.6. This value is close to the default value of 1% of the maximum flow accumulation raster (286212). The accumulation threshold has a major influence on the total number of watersheds for a particular region. The increase in the threshold number removes the lowest-order drainage segments and eliminates the small watersheds and merges them into larger watersheds. On the other hand, a decrease in the threshold will result in denser drainage network and a greater number of delineated watersheds. Therefore the definition of watershed requires a user involvement, where the user selects a threshold value to obtain the desired number and size of watersheds. However, it has been observed that for the purposes of delineating 2D slope profiles, the recommended value of 1% of the maximum flow accumulation raster performs well. At the end of the above mentioned steps, watershed polygons are created for a given DEM where the outline of each watershed is the ridge line. The delineated watersheds for the sample area can be seen in Figure 5.7.

A slope profile is a line across the ground surface largely or entirely following the direction of the true slope. In most cases, the slope profiles should be drawn between a ridge line and a drainage line in the flow direction or the steepest path of the slope surface (Luo, 2006). Ridge lines can be created using the standard watershed delineation procedure. After the initial watershed delineation procedure, two different approaches can be followed to delineate slope profiles.

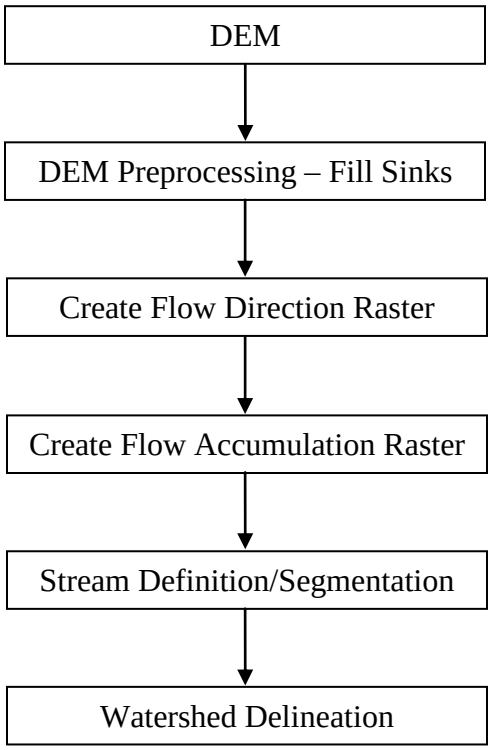


Figure 5.1 Flowchart for watershed delineation from DEM

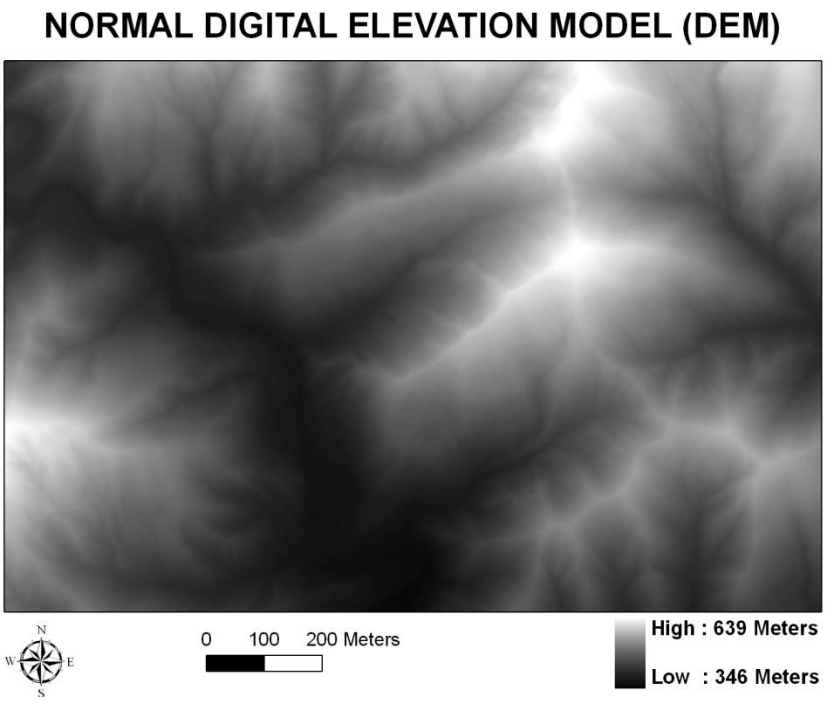


Figure 5.2 Normal digital elevation model with sinks filled

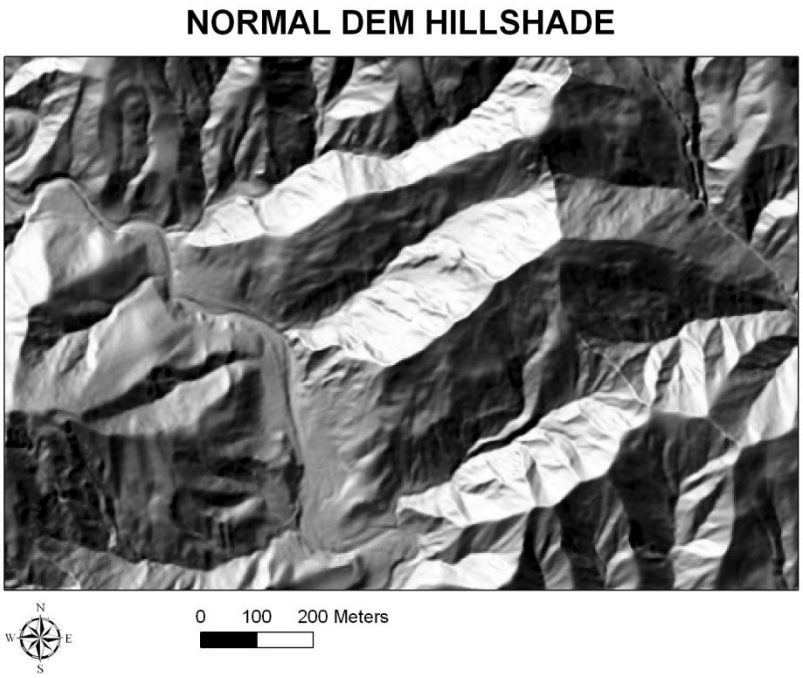


Figure 5.3 Hillshade raster of normal DEM

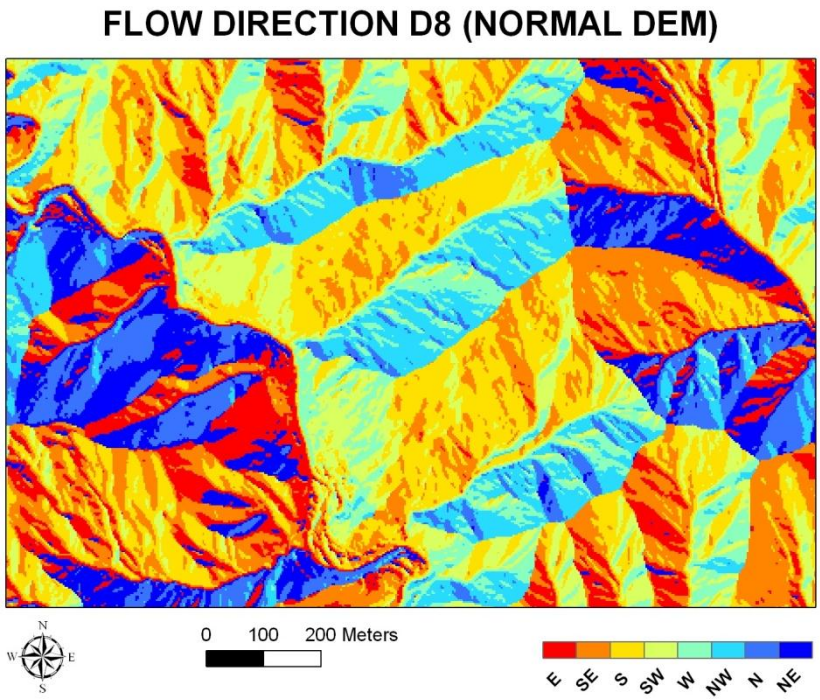


Figure 5.4 Flow direction raster

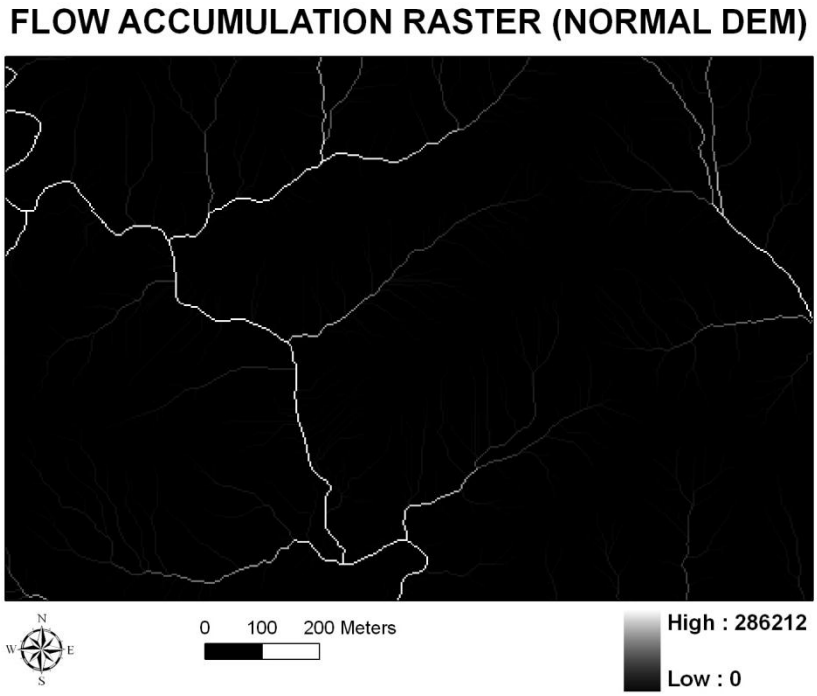


Figure 5.5 Flow Accumulation Raster

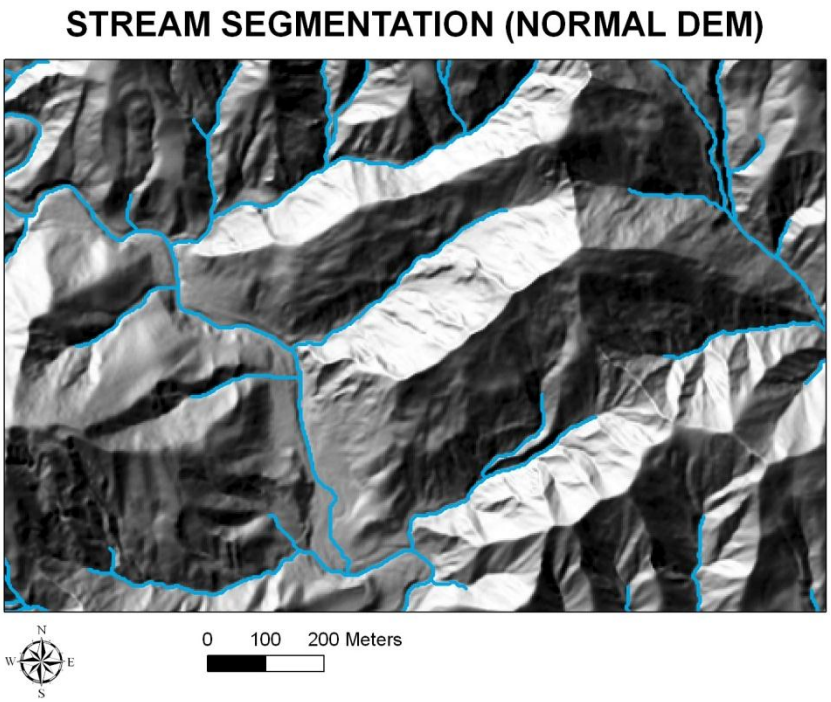


Figure 5.6 Stream segmentation

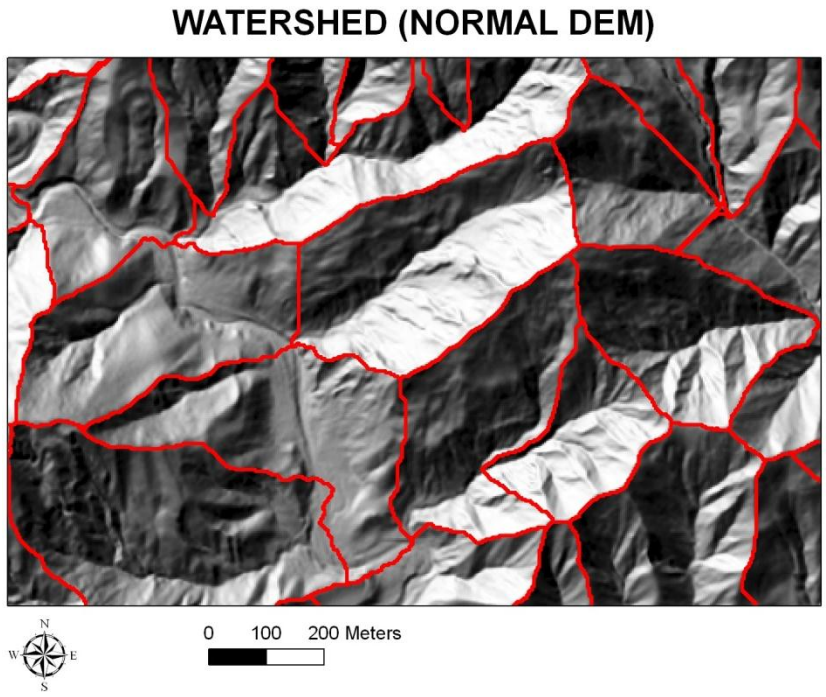


Figure 5.7 Watershed delineation

5.2.1 Slope Unit Approach

The first approach involves the delineation of slope units. A flowchart of the slope unit approach is shown in Figure 5.8. A slope unit can be defined as one slope part (right/left part) of a watershed. In addition to the normal watershed polygons delineated from normal DEM, inverse watershed polygons are delineated from an inverse DEM. Inverse DEM can be created using the raster calculator in ArcGIS under spatial analyst tools, wherein the high elevation values in normal DEM can be transformed to low values and low values can be transformed to high values. Inverse DEM is then fed to Arc Hydro and inverse watersheds are delineated in the same manner explained in the previous section. The watersheds delineated from the inverse DEM for the sample area can be seen in Figure 5.9. In Figure 5.9, inverse watersheds are shown on a hillshade raster created from inverse DEM so that the difference between the normal and inverse DEMs can be seen. Afterwards slope units are generated by dissolving and merging watersheds from both normal and inverse DEMs (see Figure 5.10). Flat areas which have slope values smaller than 10^{o} are deleted from slope unit polygons. For the next steps, a number of slope units are selected so that the rest of the procedure can be demonstrated more clearly (see Figure 5.11 for the selected slope units). Using the flow direction grid created from normal DEM and ArcGIS spatial analyst zonal statistics tool, each slope unit is assigned an average flow direction angle (see Figure 5.12). A buffer analysis is performed with a negative buffer distance (1 meter) on normal watershed polygons in order to avoid problems at the boundary of two neighboring normal watersheds. A code is written in Visual Basic, to create points in user defined intervals along the ridge lines (buffered normal watershed polygon) (see Figure 5.13) (see Appendix C, Table C.1). Each ridge point is then assigned the average flow direction angle of the slope unit which they are in. Again using a Visual Basic code, a routine is executed in which a polyline is drawn from each ridge point in the direction of the average flow direction angle of the slope unit and the polyline is extended to the border of the slope unit (see Figure 5.14) (see Appendix C,

Table C.2). Finally, slope profile polylines are converted to three dimensional polylines by interpolating elevations from normal DEM. These 3D polylines are used to calculate the geometric properties of the slope profiles, such as slope angle, height and shape. Using the slope unit approach, most of the areas inside the slope units are uniformly covered by slope profile polylines. However, the slope profile polylines are drawn using the average flow direction angle for the whole slope unit therefore they don't necessarily represent true slope.

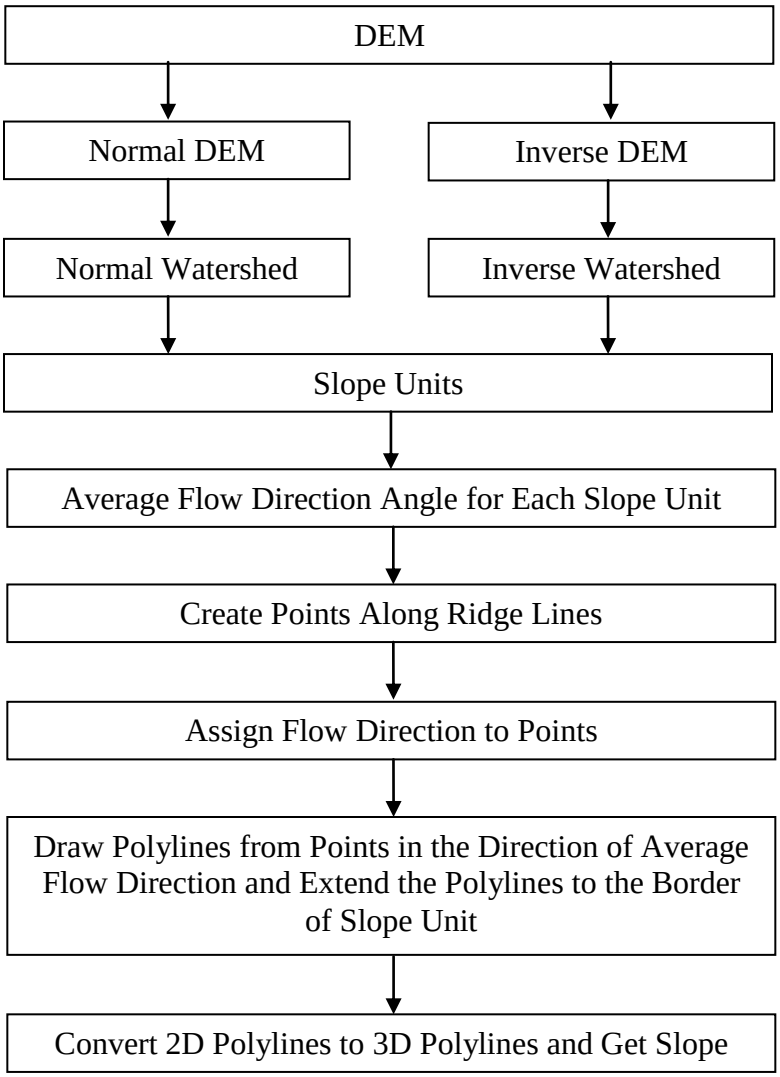


Figure 5.8 Flowchart for slope profile delineation – Slope unit approach

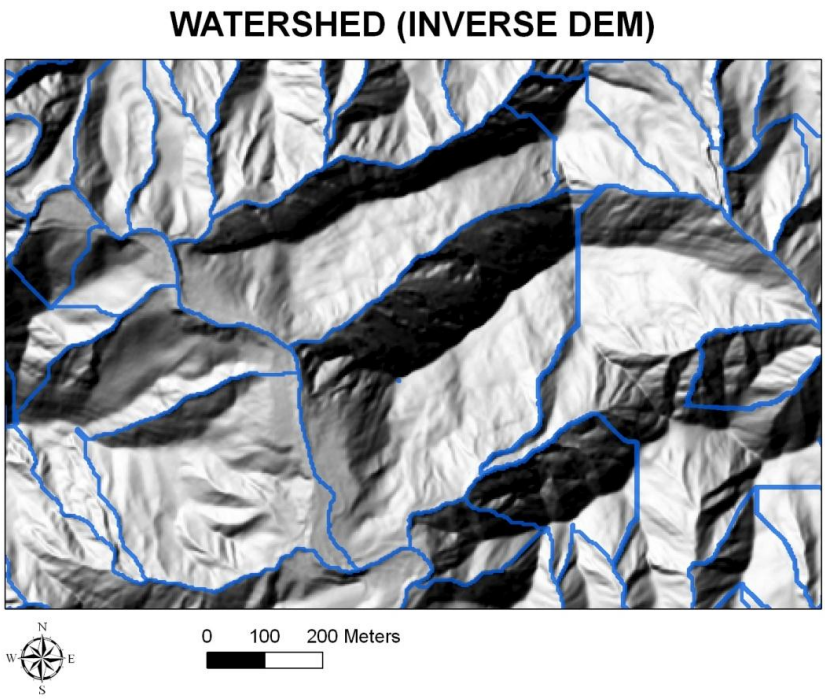


Figure 5.9 Watershed delineation from inverse DEM

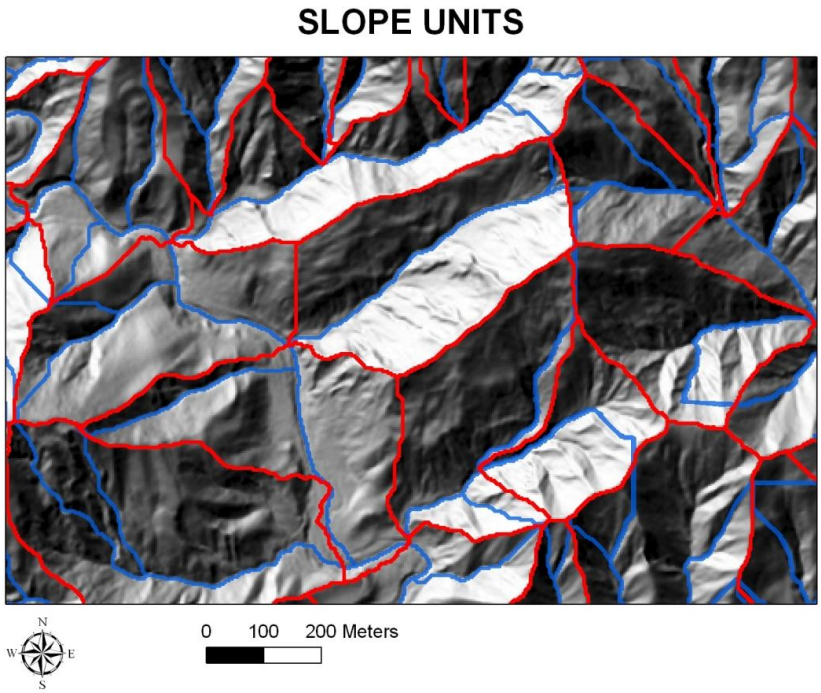


Figure 5.10 Slope unit delineation

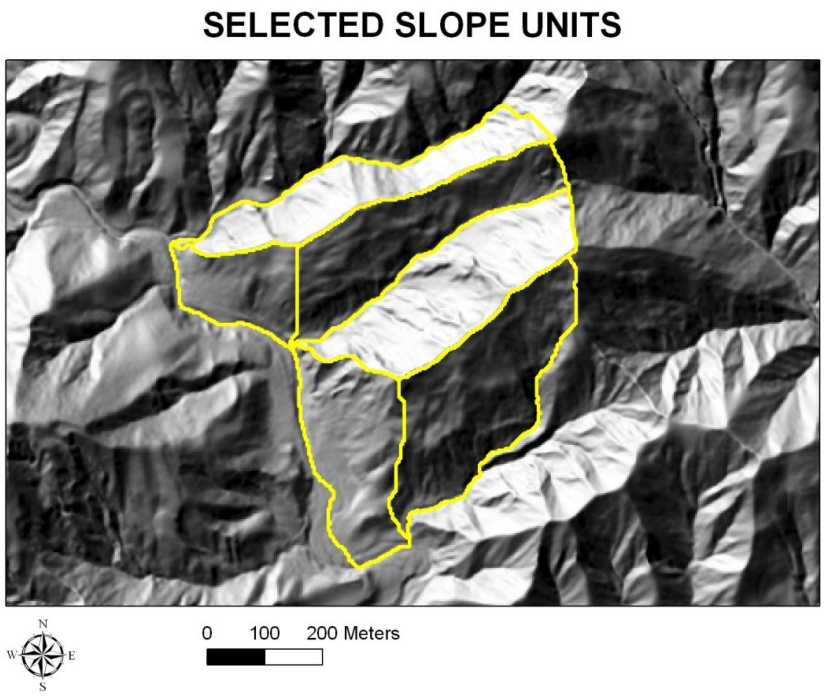


Figure 5.11 Selected slope units

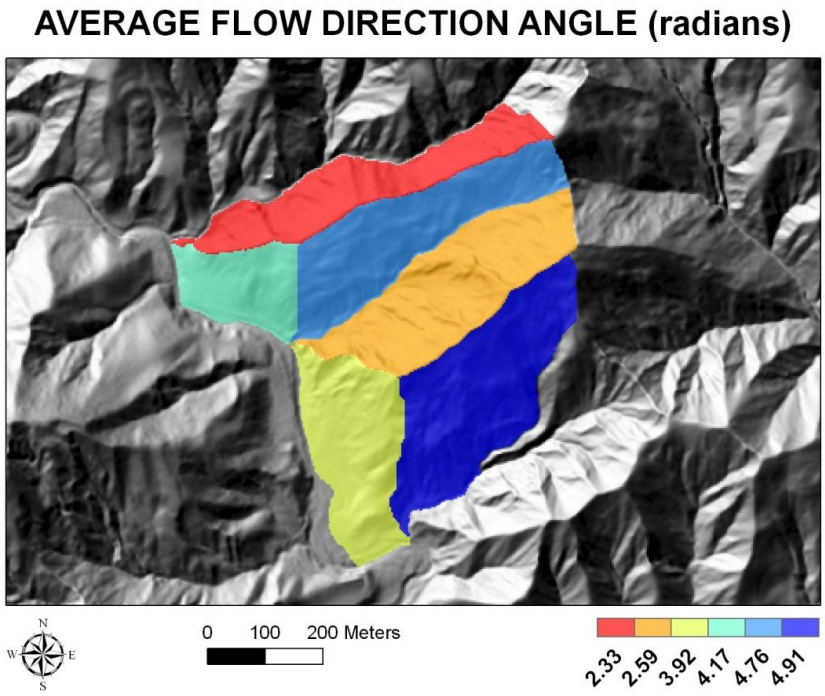


Figure 5.12 Assigning average flow direction angles to selected slope units

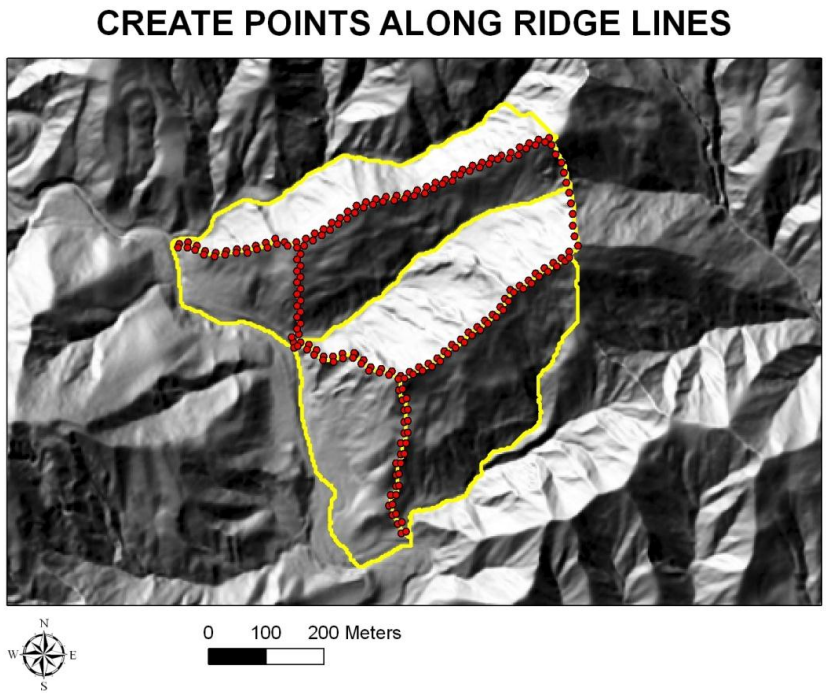


Figure 5.13 Creating points along ridge lines

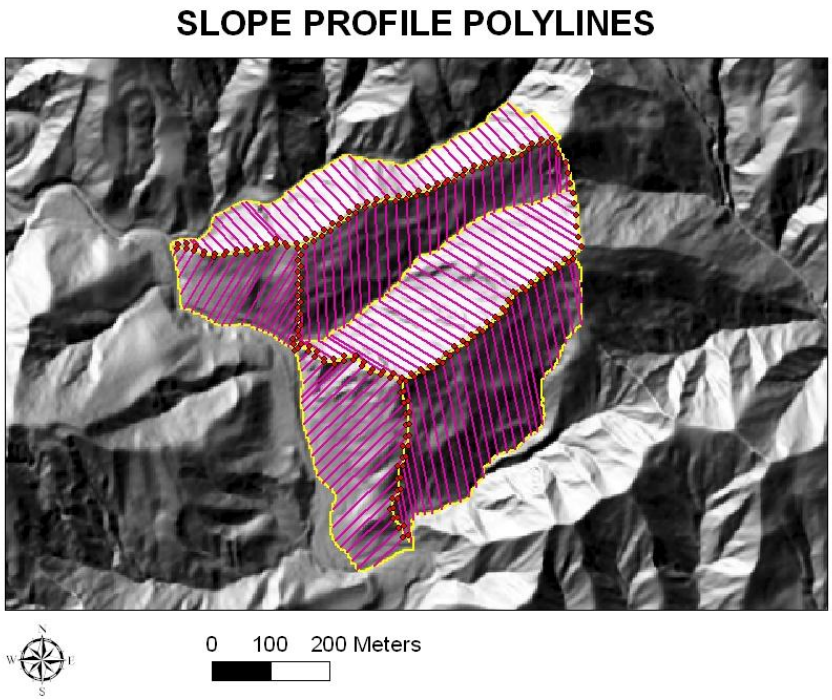


Figure 5.14 Slope profile polylines – Slope unit approach

5.2.2 Steepest Descent Approach

The second approach does not involve slope units therefore there is no need for using an inverse DEM. A flowchart of the steepest descent method is shown in Figure 5.15. In the process of delineating watersheds from normal DEM, a drainage network is also created (see Figure 5.6). This drainage network can directly be used as drainage lines when drawing slope profile polylines from ridge lines to drainage lines. Therefore, this approach uses only watersheds from normal DEM and drainage lines created during the delineation of these watersheds (see Figure 5.16). However, since no slope units are created, there are no polygons that the slope profile polylines can be drawn inside their borders. A different strategy must be adopted to create slope profile polylines. The different strategy proposed in this part involves drawing steepest descent polylines from the points along the ridge lines until they intersect a drainage line. Watershed polygon buffers are created in a similar fashion with the slope unit approach. The flat areas which have slope values smaller than 10^{o} are deleted from the watershed polygons. Ridge points are created along the watershed polygons using the same code written for the slope unit approach. A Visual Basic code is written which draws steepest descent polylines from each point until they intersect a drainage line or a flat area (see Figure 5.17) (see Appendix C, Table C.3). These polylines may or may not be straight lines depending on the topography. However, slope profile polyline is defined as a straight line. Therefore, a straight polyline is drawn from each ridge point to the point where the steepest descent polyline drawn from that point intersects a drainage line or a flat area (see Figure 5.18). Finally, slope profile polylines are converted to three dimensional polylines so that the geometric properties of the profiles such as slope angle, height and shape can be determined. Using the steepest descent approach, some of the areas inside the watersheds aren't covered because instead of being drawn in the direction of an average direction angle, the slope profile polylines follow the true slope from each point.

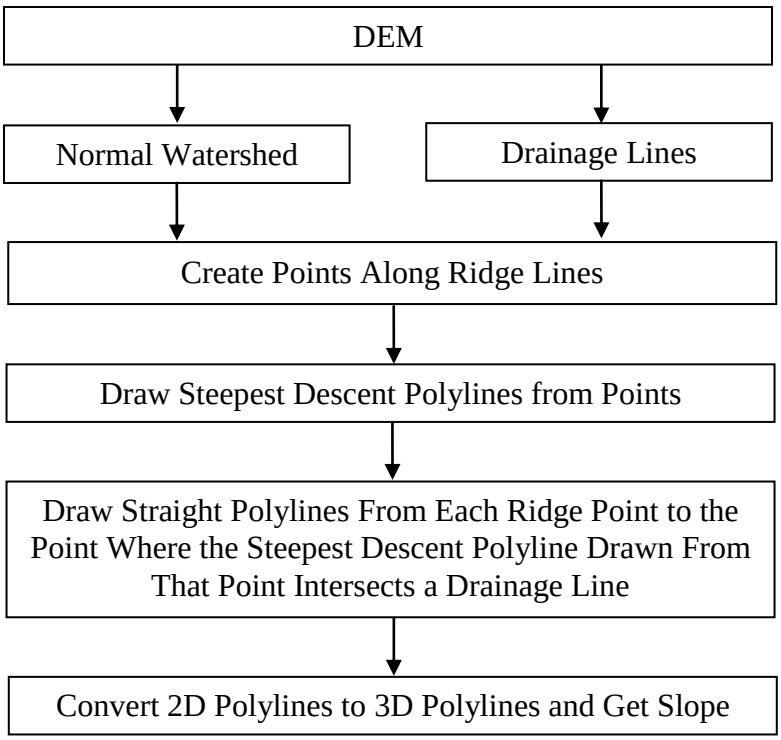


Figure 5.15 Flowchart for slope profile delineation – Steepest descent approach

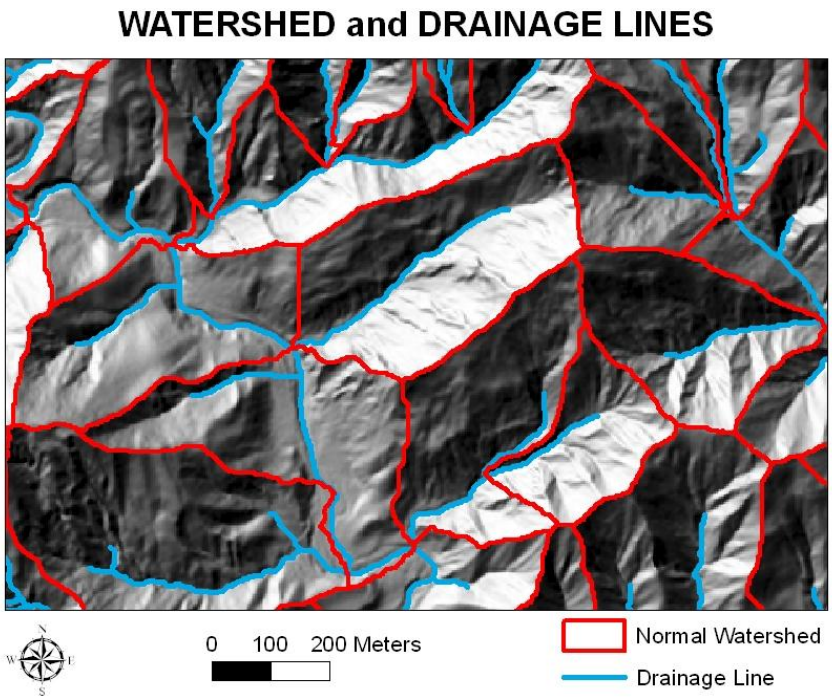


Figure 5.16 Watersheds with drainage lines

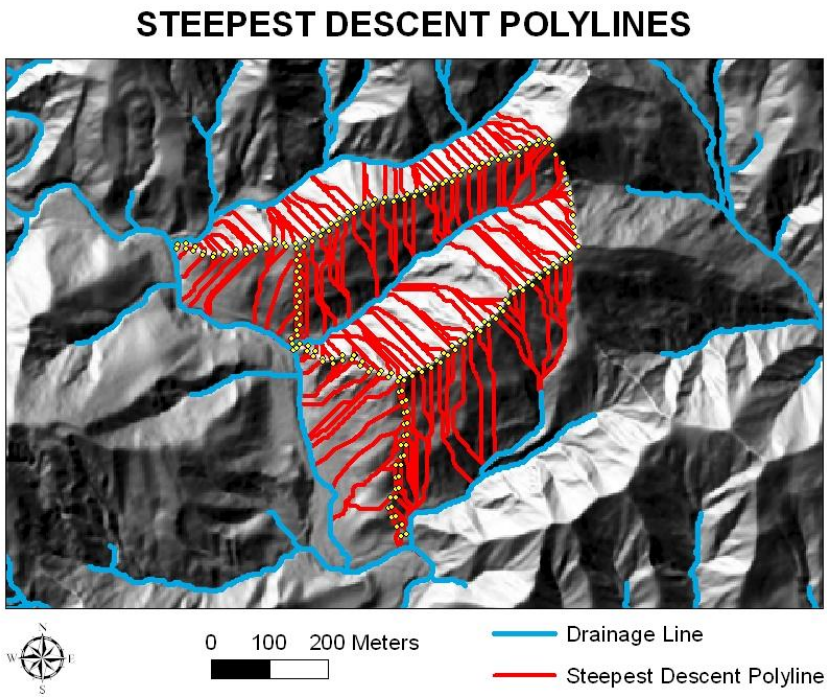


Figure 5.17 Steepest descent polylines

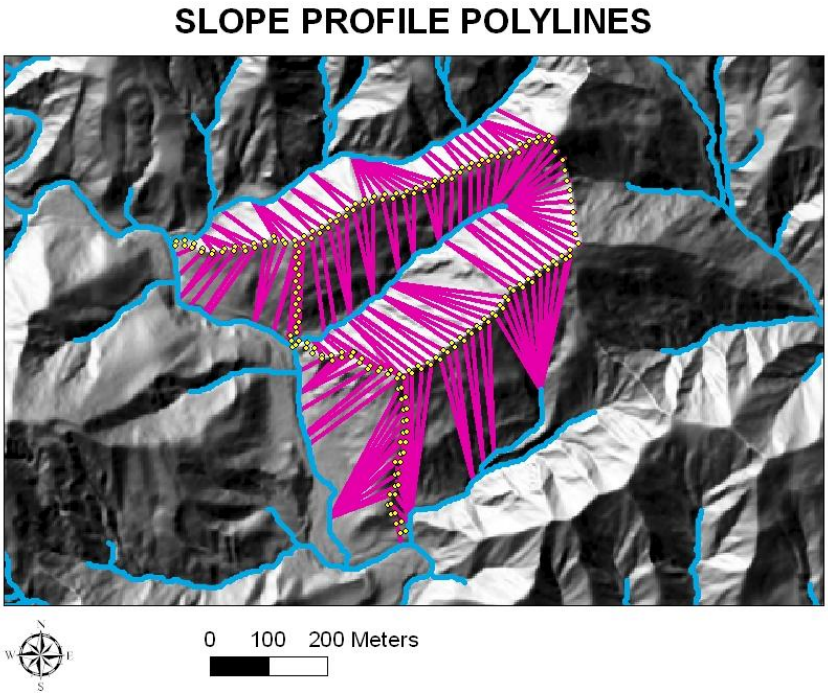


Figure 5.18 Slope profile polylines – Steepest descent approach

Both approaches use GIS-based hydrology tools to extract ridge and drainage lines. Each approach has its advantages and drawbacks. For landscapes where significant topographic irregularities exist in a given slope unit, the slope unit approach would yield a better area coverage of slope profile polylines. On the other hand, steepest descent approach may yield less area coverage because the polylines follow the direction of the true slope instead of an assigned constant direction angle. Computationally, the steepest descent method is more advantageous than the slope unit method because in the steepest descent method there is no need to delineate inverse watersheds. However, the two approaches can also be used together. The slope profiles obtained from each approach can be combined together to take into account different possible slope profiles.

Furthermore, high resolution images together with the digital elevation models can be utilized to assess the accuracy of the slope profile polylines created for a region. Integrating these various data sets allows them to be used dynamically to conduct fly-through of images of real scenes to allow the scientist to visualize key aspects of the topography. The pair of images shown in Figure 5.19 is high resolution satellite imagery and airborne LIDAR DEM of the same location. The imagery can be draped onto a DEM using ArcScene and visually explored in a 3D display environment. When digitally merged and geo-registered, the resulting 3D scene images can be used as the basis for robust hazard assessment and virtual fly-though studies. Figure 5.20 shows the 3D rendering of the sample area used in the previous sections. The pictures in Figure 5.20 are created by draping the satellite imagery over digital elevation model using ArcScene. The second picture in Figure 5.20 shows the slope profile polylines created by slope unit approach for the sample area. Visualizing the polylines in 3D rendering images provides a better understanding of the orientation and direction of the slope profile polylines. Using 3D fly-through capabilities of ArcScene, the slope profile polylines created for a region can be inspected in an efficient manner.

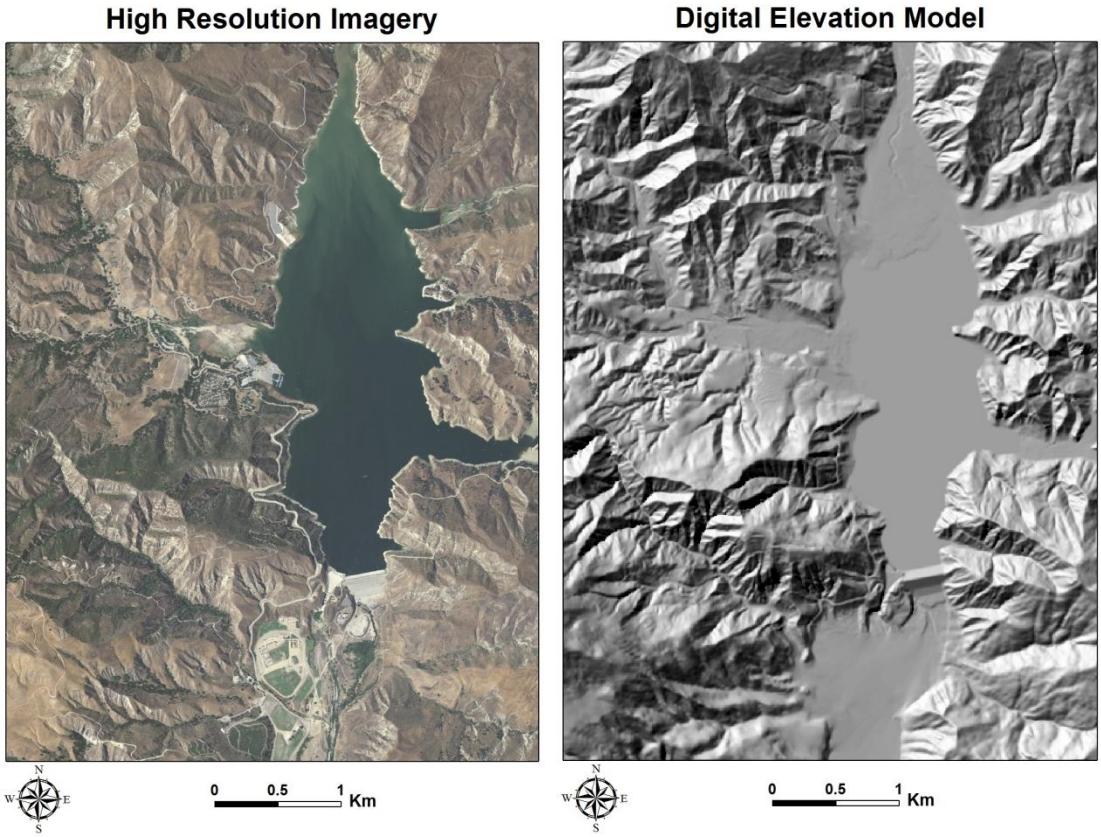


Figure 5.19 High resolution satellite imagery and DEM of same region

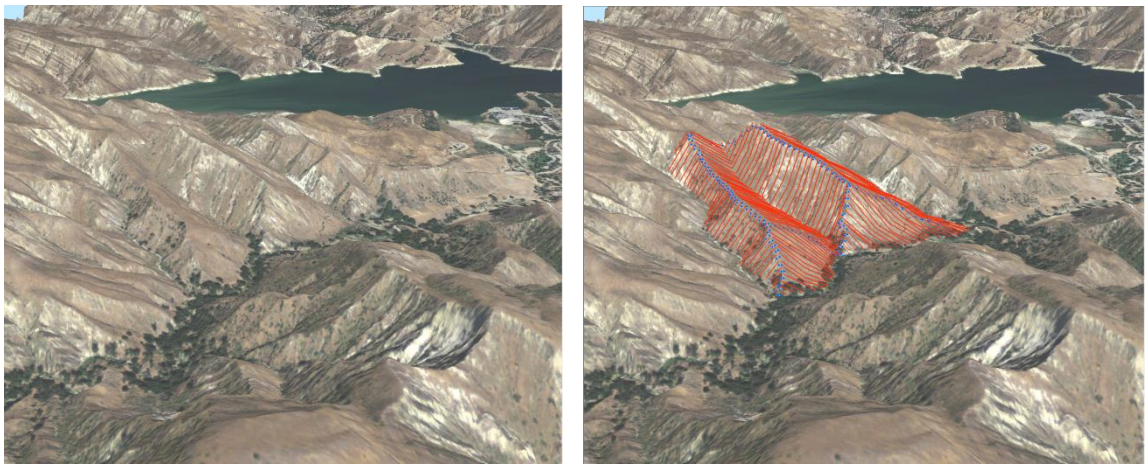


Figure 5.20 Inspecting slope profile polylines in ArcScene using 3D rendering

5.3 Delineation of Two Dimensional Slope Profiles from Digital Elevation Models along Highways

Landslides along highway sections may cause loss of life, damage to vehicles and destruction of the roadway infrastructure. It is desirable to produce landslide susceptibility maps along highways for a regional hazard analysis. In this section, an extension to the approaches presented in the previous section that can be used specifically to delineate slope profiles along highway sections is presented.

Highway sections can be considered as one of three different categories. The highway can be on a ridge line or a drainage line or it can be at an elevation between a ridge line and a drainage line. Examples of slope failures along highways sections for all three categories can be seen in Figure 5.21. If the highway section is on a ridge line or a drainage line, tools developed in the previous section can directly be used to delineate the 2D slope profiles on both sides of the highway section without modification. However, if the highway section is in between ridge and drainage lines, it requires a different approach to delineate the profiles for both the up-slope and down-slope components.

Figure 5.22 and 5.23 shows the delineation of slope profiles in the case where the highway section is on a ridge line and a drainage line. A hillshade created from a high resolution DEM (NED 1/9 Arc-Second) and the road network can be seen in Figure 5.22 (a). By using Arc Hydro tools watersheds and drainage lines are delineated from the normal DEM (See Figure 5.22 (b)). As can be seen from Figure 5.22 (a) and 5.22 (b) the highway section going through the valley is overlapping with a drainage line and the highway section on the top of the left side of the valley is overlapping with a ridge line. Therefore, points can be created along this highway section and using the steepest descent approach, polylines can be drawn between the drainage lines and the ridge lines and slope profiles can be created accordingly. Figure 5.23 (a) shows sample slope profile polylines (yellow line) drawn on a high resolution satellite image draped over the high resolution DEM. Figure 5.23 (b) shows the slope profile for the case where highway

section is on a ridge line. Figure 5.23 (c) shows the slope profile for the case where highway section is on a drainage line.

In the case where the highway section is located at an elevation between a ridge line and drainage line, the stabilities of both the up-slope and down-slope sections should be analyzed. Similar to creating points along the ridge lines, in this case the points can be created along the highway sections at user defined intervals. The Visual Basic code that was used to create points along ridge lines can be used to create points along highway sections with a slight modification. From these points, steepest descent polylines can be drawn until they reach a drainage line. In the same manner steepest ascent polylines can be drawn from these points until they reach a ridge line. Steepest ascent polylines can be drawn using the same code written to create steepest descent polylines but using an inverse DEM instead of a normal DEM. Figure 5.24 and 5.25 shows the delineation of slope profiles in the case where the highway section is at between a ridge line and a drainage line. The hillshade and the road network can be seen in Figure 5.24 (a). Watersheds and drainage lines that are delineated from the normal DEM by using Arc Hydro tools can be seen in Figure 5.24 (b). Figure 5.25 (a) shows sample slope profile polylines drawn for both up-slope and down slope on a high resolution satellite image draped over the high resolution DEM. Figure 5.25 (b) shows the up-slope profile and Figure 5.25 (c) shows the down-slope profile.

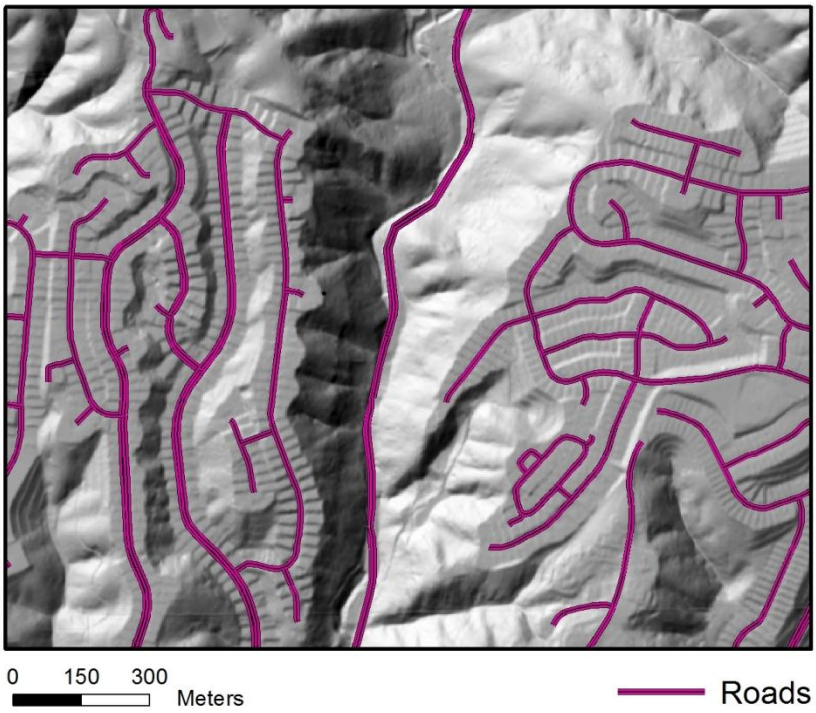
a)

b)

c)

Figure 5.21 Examples of slope failures along highway sections (a) Highway section on a drainage Line (b) Highway section on a ridge line (c) Highway section at an elevation between a ridge line and a drainage line (Pictures taken by Dr. David Frost)

a)



b)

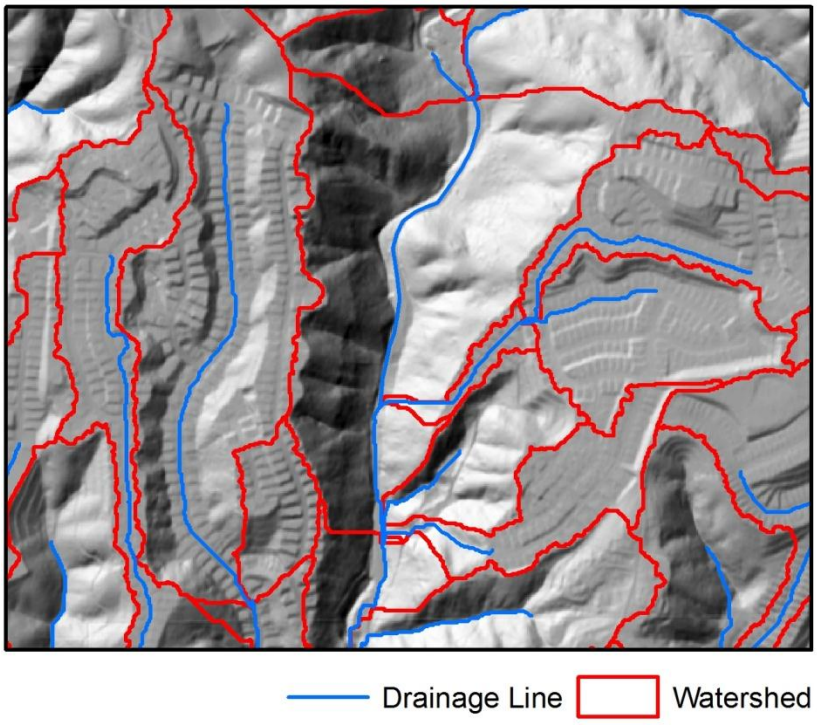
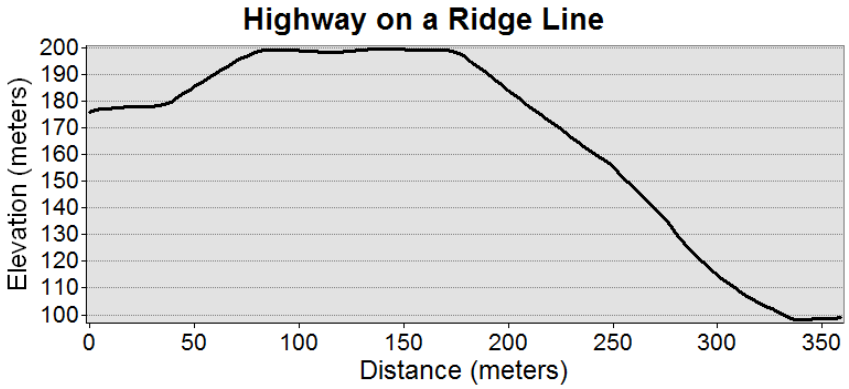


Figure 5.22 Delineation of slope profiles along highway sections for the cases when highway section is on a ridge line and drainage line (a) Hillshade and the road network (b) Watersheds and drainage lines delineated from the normal DEM

a)

b)



c)

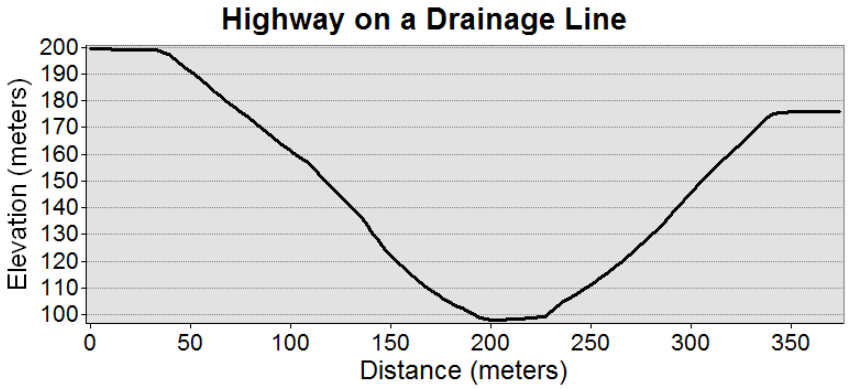
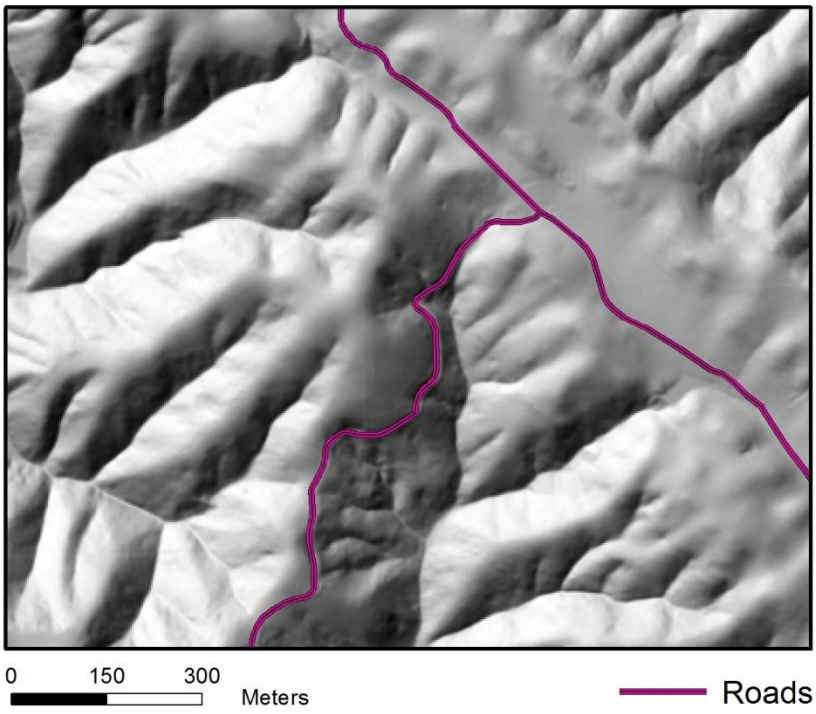


Figure 5.23 Delineation of slope profiles along highway sections for the cases when highway section is on a ridge line and drainage line (a) 3D rendering of the scene (b) Slope profile for the highway section on the ridge line (c) Slope profile for the highway section on the drainage line

a)



b)

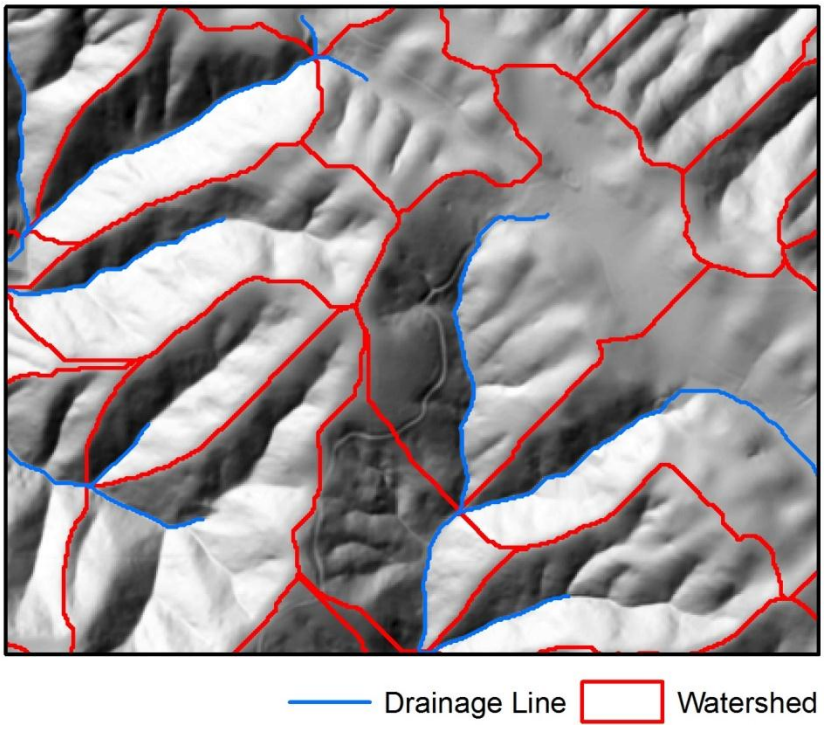
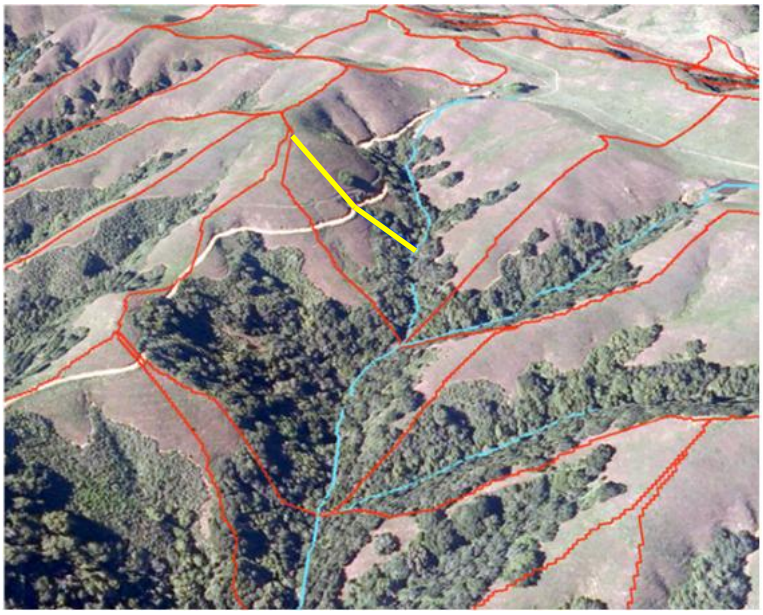
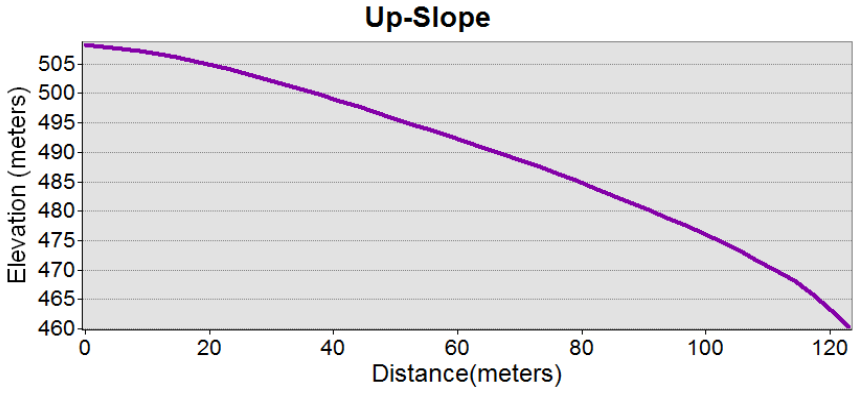


Figure 5.24 Delineation of slope profiles along highway sections for the case when highway section is between a ridge line and drainage line (a) Hillshade and the road network (b) Watersheds and drainage lines delineated from the normal DEM

a)



b)



c)

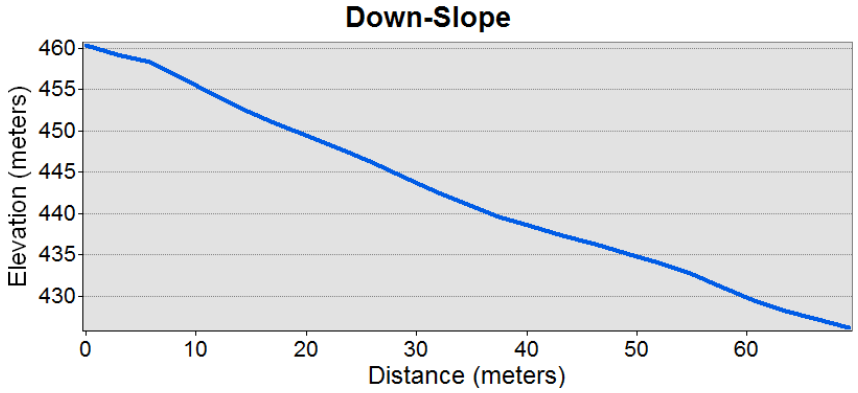


Figure 5.25 Delineation of slope profiles along highway sections for the cases when highway section is between a ridge line and drainage line (a) 3D rendering of the scene (b) Slope profile for the up-slope (c) Slope profile for the down-slope

5.4 Summary

GIS-based regional landslide hazard analyses involve the preliminary selection of a suitable mapping unit. There are readily available methods for the extraction of attributes such as slope for grid based analyses. These tools are offered by standard GIS platforms.

In the case of coherent landslides, grid based methods which utilize the infinite slope model don't provide reliable results. Analytical methods for the evaluation of static and dynamic responses of slopes require the two dimensional profile of the slope.

Alternative methodologies to delineate two dimensional slope profiles from a given Digital Elevation Model (DEM) are explored. Two different approaches which use the readily available GIS-based hydrology tools are presented for delineation of two dimensional slope profiles.

The slope unit approach involves the delineation of slope units using normal and inverse DEMs. After the slope units are delineated, average flow direction angles are assigned for each slope unit. Next, polylines are drawn from the points created along ridge lines in the direction of the average flow direction and these lines are extended to the border of the slope unit.

Steepest descent approach does not involve the delineation of slope units. In this second approach, only normal watersheds and corresponding drainage lines are delineated using Arc Hydro tools. Steepest descent polylines are drawn from the points created along ridge lines until they intersect the drainage lines or the flat areas. These polylines are not always straight lines. Therefore, a straight polyline is drawn from each ridge point to the intersection point.

An extension to these approaches that can be used specifically to delineate slope profiles along highway sections is also presented.

CHAPTER 6

SYTEM DEVELOPMENT AND PILOT STUDY

6.1 Introduction

An integrated framework is developed for combining the metamodels described in Chapter 3 for predicting the static factor of safety and in Chapter 4 for predicting the performance of slopes under dynamic loading with the automated slope profile delineation method introduced in Chapter 5. Due to the spatial nature of earthquake triggered landslides, geographic information system technology (GIS) plays an important role in the integration of soft computing methods to regional landslide hazard analysis.

The integrated framework is evaluated using a case study of the 1989 Loma Prieta earthquake (Mw = 6.9) within the 7.5 min. Laurel, CA quadrangle. The motivation for choosing a 7.5 min. quadrangle is to focus the scope of the case study to a region that had a number of important attributes. There are only a few landslide inventories available from historical earthquakes and many of these inventories do not provide detailed information regarding the different types of landslides. The landslide inventory for Loma Prieta earthquake includes detailed information about the types of the landslides; moreover the database provides the displacement values for several coherent deep-seated landslides. From the earthquakes which have comprehensive landslide inventories, the Loma Prieta earthquake triggered a greater proportion of large coherent landslides than Chi-Chi and Northridge earthquakes (Khazai and Sitar, 2004). Laurel quadrangle is chosen because it contains the highest concentration of coherent landslides from the event (Keefer and Manson, 1998). A seismic landslide hazard analysis is also performed for the same region for a future scenario earthquake (Mw = 7.03) on the San Andreas Fault.

6.2 Coherent Landslides Triggered by the Loma Prieta Earthquake

The 1989 Loma Prieta earthquake (M_w=6.9) triggered thousands of landslides. Most of these landslides occurred within an area of approximately 2,000 km^2 in the rugged, heavily vegetated southern Santa Cruz Mountains through which the rupture zone passes (Keefer, 1998). Landslides in this region damaged at least 200 residences, many roads, and numerous other structures. Similar to the other earthquakes worldwide, the most common types of landslides were rock falls, rock slides, and disrupted soil slides. However, deep seated and coherent slides were also moderately common. The shallow disrupted landslides were most common in weakly to moderately cemented sedimentary rocks. On the other hand, the coherent landslides were most common in preexisting landslide deposits and artificial fill (Keefer and Manson, 1998). The earthquake occurred during the fourth year of drought conditions and at the end of a dry summer season, therefore the ground conditions were especially dry.

One of the most important slope-failure effects of the Loma Prieta earthquake was the generation of numerous large, deep-seated landslides that occurred in and around the Summit Ridge area near the rupture zone (Figure 6.1). A general reconnaissance of more than 50 large earthquake triggered coherent landslides was performed after the earthquake. Out of these landslides, 20 were selected for more rigorous inspection (Keefer et al., 1998). A list of 20 landslides and their characteristics can be seen in Table 6.1. These landslides were as much as 980 m long and 1,300 m wide. Maximum landslide depths ranged from 27 to possibly more than 100 m (Cole et al., 1998). Main scarps and heads were well developed, and measured landslide displacements ranged from 25 to 244 cm. All of the landslides exhibited geomorphic indications of previous landslide movement. In addition to deep-seated landslides, the earthquake opened ground cracks at hundreds of localities throughout the Santa Cruz Mountains. Keefer and Manson (1998) states that many of these ground cracks marked incipient landslides, which would have developed more fully had the ground conditions been wetter or the

earthquake shaking more severe. Ground cracks interpreted as incipient landslides mostly occurred along or near ridge crests and upslope from preexisting landslide deposits.

The coherent deep-seated landslides in the Summit Ridge area presented an opportunity for the researchers to evaluate existing methods for analyzing slope displacements under earthquake loading (Keefer, 1991; Cole et al., 1998; Strenk, 2010) . Cole et al. (1998) studied two reactivated landslides (Lower Schultheis Road West and Ditullio landslides), using subsurface logging and sampling, and laboratory testing. They performed both pseudostatic and Newmark displacement analysis and compared the actual landslide behaviors with analytical models. Cole et al. (1998) reported that the laboratory strengths calculated for the landslide materials were too high to accurately predict stability and displacements; therefore they used backcalculated shear strengths to improve the accuracy of calculated displacements. Keefer (1991) investigated the Upper Schultheis Road and Villa Del Monte landslides. They also backcalculated the strength of the slope materials and used these backcalculated strength values to estimate the stability of the slopes under higher ground-water conditions and to calculate the displacements that could occur in future earthquakes. Strenk (2010) studied the Upper Laurel and Ditullio landslides, using probabilistic seismic slope stability analysis and compared the real slope displacements with the displacement results obtained from 20 different deformation based methods including simple rigid block methods to more complicated decoupled and coupled methods.

In the following sections, the Lower Schultheis Road East landslide (see Figure 6.1) will be analyzed in detail using dynamic FLAC analyses. The performance of the SVM metamodel (classifier) created in Chapter 4 using both artificial and real earthquake records will be investigated by comparing the class calculated using FLAC analysis and the class predicted using the SVM classifier.

Table 6.1 Characteristics of large landslides in Summit Ridge area (n.d. = not determined, * = Investigated in detail in this study) (adopted from Keefer et al. (1998))

	Landslide name	Length (m)	Width (m)	Slope (^{o})	Displace-ment (m)	Predominant Geologic Unit(s)
1	Majestic Drive	420	170	20	n.d.	Tb
2	Old Santa Cruz Highway	670	1300	15-20	n.d.	Tv
3	Upper Schultheis Road	460	390	15	2.44	Tb,Tv
4	Ralls Drive	910	730	15	1.02	Tb,Tv
5	Villa Del Monte	980	870	12-15	.53	Tsr,Tst,Tb
6	Taylor Gulch	210	270	16	.53	Tsr
7	Upper Morrell Road	270	550	15	2.01	Tsr
8	Lower Morrell Road	460	670	15	n.d.	Tb,Tbm
9	Burrell	230	460	15-20	.71	Tbm
10	Upper Redwood Lodge Road	820	640	15-20	.25	Tsr,Tv
11	Long Branch	120	180	22	.46	Tv
12	Stetson Road	200	460	20	.61	Tv,Tsr
13	Amaya Ridge	120	240	22	1.00	Tp
14	Hester Creek North	150	90	25	.79	Tp
15	Hester Creek South	180	180	25-30	.25	Tp
16	Lower Redwood Lodge Road	180	280	20-25	n.d.	Tsr
17	Lower Schultheis Road East*	300	120	20	.69	Tsr
18	Lower Schultheis Road West	110	110	15-20	.65	Tsr
19	Soquel-San Jose Road	170	170	n.d.	.46	Tp
20	Cornstock Mill Road	820	150	n.d.	.65	Tsr

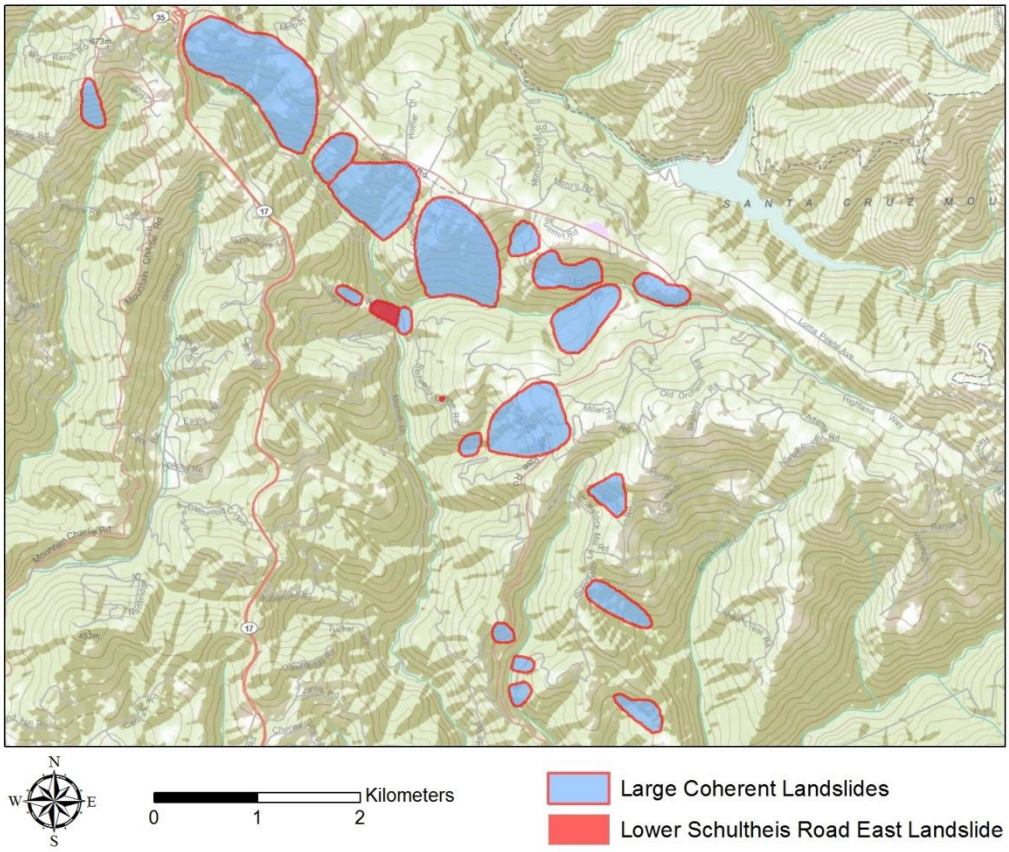


Figure 6.1 Large coherent landslides and Lower Schultheis East landslide in Summit Ridge area after Loma Prieta earthquake

6.2.1 Simulation of Loma Prieta Earthquake Ground Motions

1989 Loma Prieta earthquake affected a region that lies along the plate boundary between the North American and Pacific plates. Many active and potentially active faults like the San Andreas and Hayward faults are included in this region. The Loma Prieta earthquake occurred at 5:04 p.m. on October 17, 1989. It had a Richter surface wave magnitude (M_S) of 7.1 and a moment magnitude (M_w) of 6.9 (Keefer and Manson, 1998). The hypocenter was located at approximately 18 km. depth and 37°02′ N., 121°53′ W., in the Santa Cruz Mountains. The fault rupture models showed that the earthquake ruptured

a 40 km long segment of either San Andreas or a nearby fault (Wald et al., 1991; Keefer and Manson, 1998).

The intensity and frequency of ground motion at a particular location is influenced by several factors such as the magnitude of the earthquake, distance and azimuth from the rupture plane, source and rupture mechanism of the earthquake, geotechnical site conditions and topographic amplification. It is not an easy task to select appropriate ground motion records for the seismic stability analysis of slopes because most of the landslides in the Summit Ridge region are at relatively large distances from actual ground motion stations that recorded the earthquake. When analyzing the landslides in the Summit Ridge region, Cole et al. (1998) used the scaled ground motion recordings at Corralitos (CLS) and Los Gatos Presentation Center (LGPC) stations at distances 16.4 km and 7 km away from the landslides. Further, Cole et al. (1998) stated that the recorder at the LGPC station was not firmly anchored to the floor and seismologists have cautioned that the data from this station may not be reliable. Keefer (1991) also used the ground motion recordings at CLS station for the two case histories. On the other hand, Strenk (2010) used attenuation relationships to estimate the response spectra at the landslide locations and matched the motion recordings from close by stations using a spectral matching algorithm to obtain spectra compatible acceleration time histories. All these case histories are for specific landslides at particular locations. However, for a regional landslide hazard analysis which involves larger areas, it is more convenient to use artificial ground motions generated from stochastic models of the earthquake (Miles and Ho, 1999).

Miles and Ho (1999) used a point source stochastic model (SMSIM) for simulating earthquake ground motions to use in a regional shallow landslide hazard analysis. However, they also stated in their paper that the stochastic point source models are not reliable for locations near the rupture plane. Despite the popularity of the point-source models, currently there is a shift towards extended (finite-fault) source models

because the point source models are unable to characterize the key features of ground motions from large earthquakes (Beresnev and Atkinson, 2002; Motazedian and Atkinson, 2005). The dependence of amplitude and duration on the azimuth to the observation point (source directivity) cannot be modeled using a point source model. Finite-fault models also affect the shape of the spectra of the seismic waves (Beresnev and Atkinson, 2002). Therefore, finite-fault modeling has been an important tool for prediction of ground motions near the epicenters of large earthquakes. In the finite-fault model, the plane of rupture is divided into subfaults, and the radiation from a large earthquake is simulated as the sum of contributions from all subfaults which act as an independent source. There are several computer codes available for stochastic finite-fault models. Two of the most commonly used computer codes for finite-fault modeling are FINSIM (Beresnev and Atkinson, 1998) and EXSIM (Motazedian and Atkinson, 2005). The FINSIM code and the input parameters for Loma Prieta earthquake were obtained by contacting the author (Beresnev, personal communication). EXSIM code is publically available but the input parameters for Loma Prieta earthquake were acquired from the author (Motazedian, personal communication). Both codes are used to simulate the earthquake motions at selected stations and the results are compared with the recorded earthquake motions. EXSIM code was found to be better at simulating the ground motions at the selected stations. There are several reasons for the better performance of the EXSIM code. Some of the reasons can be stated as follows; the EXSIM code is based on a "dynamic corner frequency" and unlike FINSIM the radiated energy is less dependent on subfault size (Motazedian and Atkinson, 2005; Assatourians and Atkinson, 2010).

The rupture plane and the epicenter for the Loma Prieta earthquake and the ground motion recording stations that were used to validate the results from EXSIM simulations can be seen in Figure 6.2. The rupture model is adopted from Wald et al.

(1991). The input modeling parameters calibrated for Loma Prieta earthquake for use with EXSIM can be seen in Table 6.2.

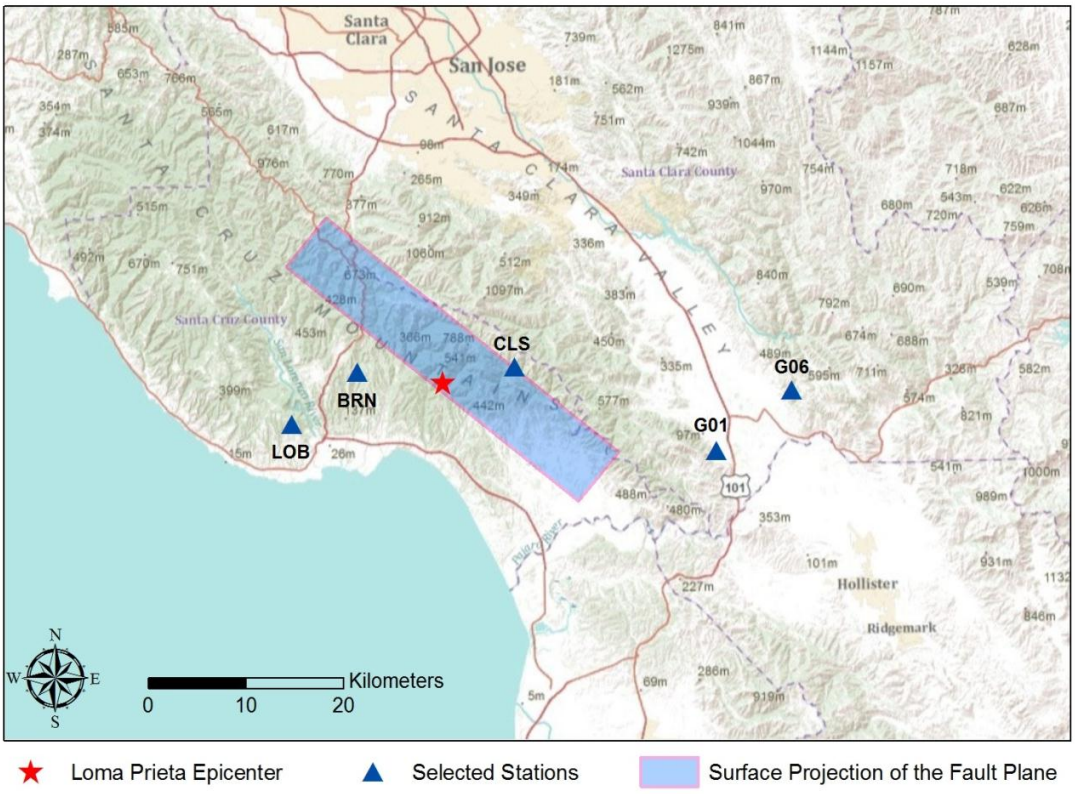


Figure 6.2 Surface projection of the rupture plane and selected ground motion stations

Table 6.2 EXSIM modeling parameters for Loma Prieta earthquake

- Fault Orientation: Strike 128^o , Dip 70^o
- Depth of top: 1.5 km
- Fault dimensions: Length 40km, Width 20 km
- Number of subfaults: 8 lengthwise, 4 widthwise
- Geometrical Attenuation:
  $R < 40km : R^{-1}$
  $40km \leq R : R^{-0.5}$
- $Q(f) : 187f^{0.56}$
- Crustal shear-wave velocity: 3.7 km/sec
- Crustal Density: 2.8 g/cm^3
- Stress drop (bars): 60
- Site amplification: generic rock (Boore and Joyner, 1997)
- Kappa: 0.030
- Pulsing area percentage: 25%
- Windowing function: Saragoni-Hart

In order to avoid complications related to the effects of local soil conditions on the earthquake motions, only the stations on sites categorized as rock were used (Beresnev and Atkinson, 2002). These include Geo-matrix Classes A, or typical shear-wave velocity of about 600 m/sec averaged over the top 30m. A list of the stations used can be seen in Table 6.3.

Figures 6.3 to 6.5 show the 5% damped pseudo acceleration spectrums of EXSIM simulated earthquake motions versus the fault normal (000) and fault parallel (090) spectrums of the recorded earthquake motions. As can be seen in these figures, EXSIM satisfactorily simulates the earthquake motions at these selected stations. Another useful property of the EXSIM code is the simplicity of entering the spatial information for the desired locations. The fault rupture location (including size, strike and dip information) is directly entered into the code with its corresponding geographic coordinate information. After the model is created, artificial earthquake motions can be obtained by just entering the latitude and longitude of the desired location (assuming generic rock site conditions). This property makes the integration of EXSIM code into a regional landslide hazard analysis using GIS very effective. Using this stochastic finite-fault model, both historical earthquakes and future scenario earthquakes can be modeled and reliable artificial ground motions can be created for regional seismic hazard analysis including the regions in close proximity to the rupture planes and epicenters.

Summit Ridge area where numerous deep-seated coherent landslides were observed after the Loma Prieta earthquake is on the fault rupture plane and in very close proximity to the earthquake epicenter. Also the region selected for the pilot study (Laurel Quadrangle) is on and around the rupture plane. Therefore, the ground motion records used in the analysis of landslides triggered by the Loma Prieta earthquake are generated using EXSIM finite-fault model with the input model parameters shown in Table 6.2. EXSIM finite-fault model is also used in this study to create the artificial earthquake records from a scenario earthquake on San Andreas Fault.

Table 6.3 Ground motion stations used for validation of EXSIM

Station Name	Distance to Hypocenter (km)	Joyner-Boore Distance (km)	Source to Site Azimuth (deg)	Geo-matrix Classification
Corralitos (CLS)	18.98	0.15	-90	A
BRAN (BRN)	19.66	3.85	90	A
Gilroy Array #1 (G01)	33.55	8.84	-44	A
Gilroy Array #6 (G06)	39.54	17.92	-60	A
UCSC Lick Observatory (LOB)	23.93	12.04	90	A

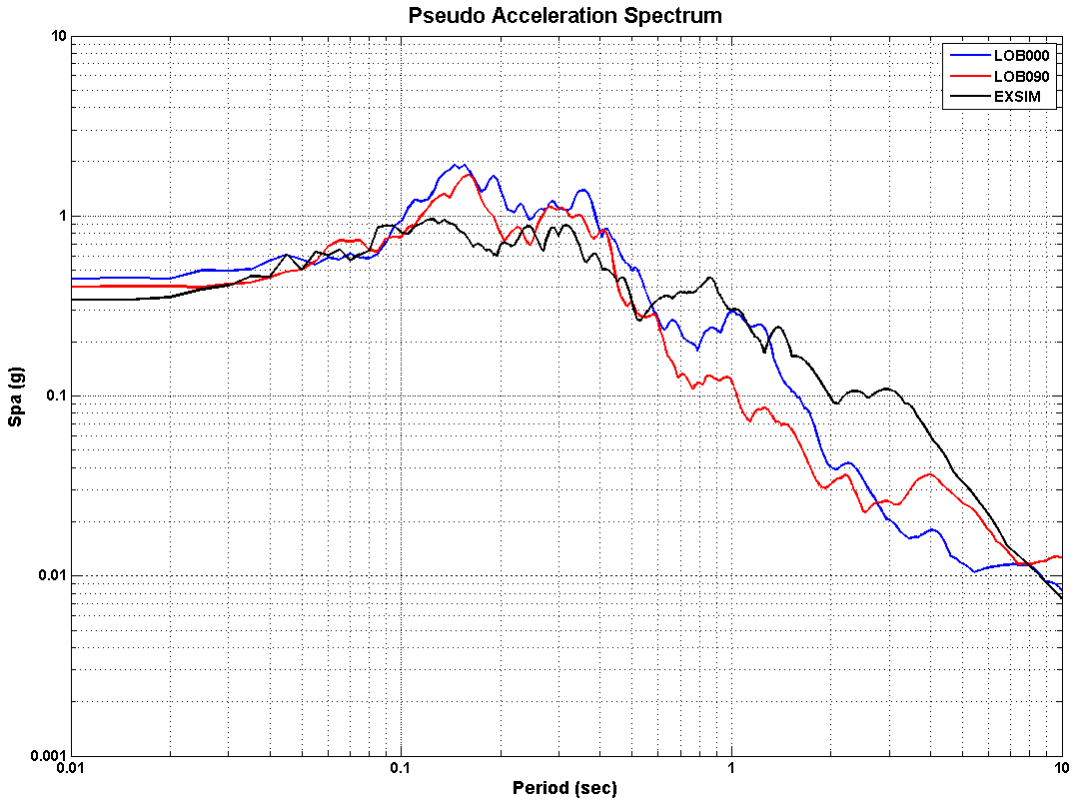


Figure 6.3 Pseudo acceleration spectrums for real and EXSIM simulated motions at UCSC Lick Observatory (LOB) station

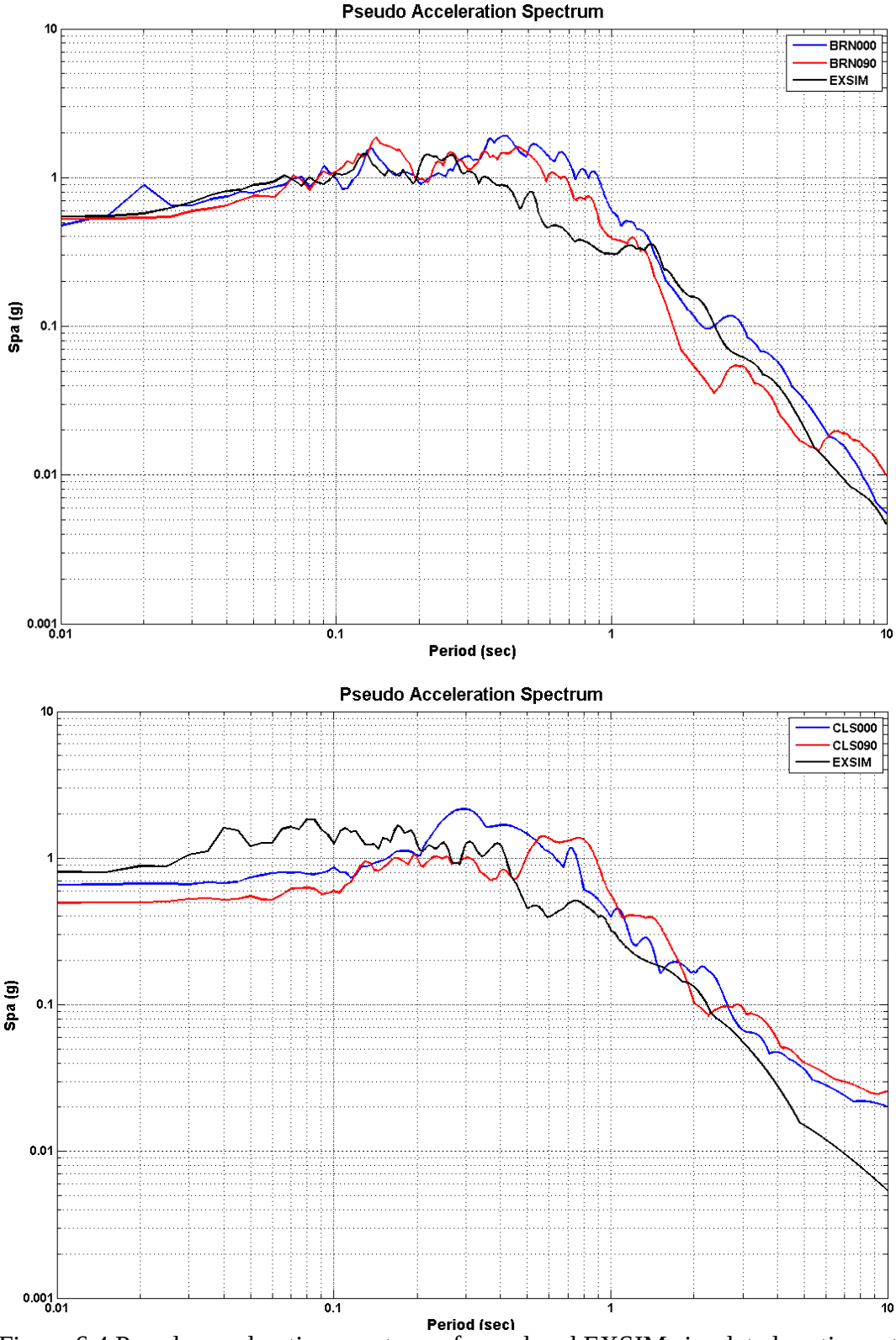


Figure 6.4 Pseudo acceleration spectrums for real and EXSIM simulated motions at BRAN (BRN) and Corralitos (CLS) stations

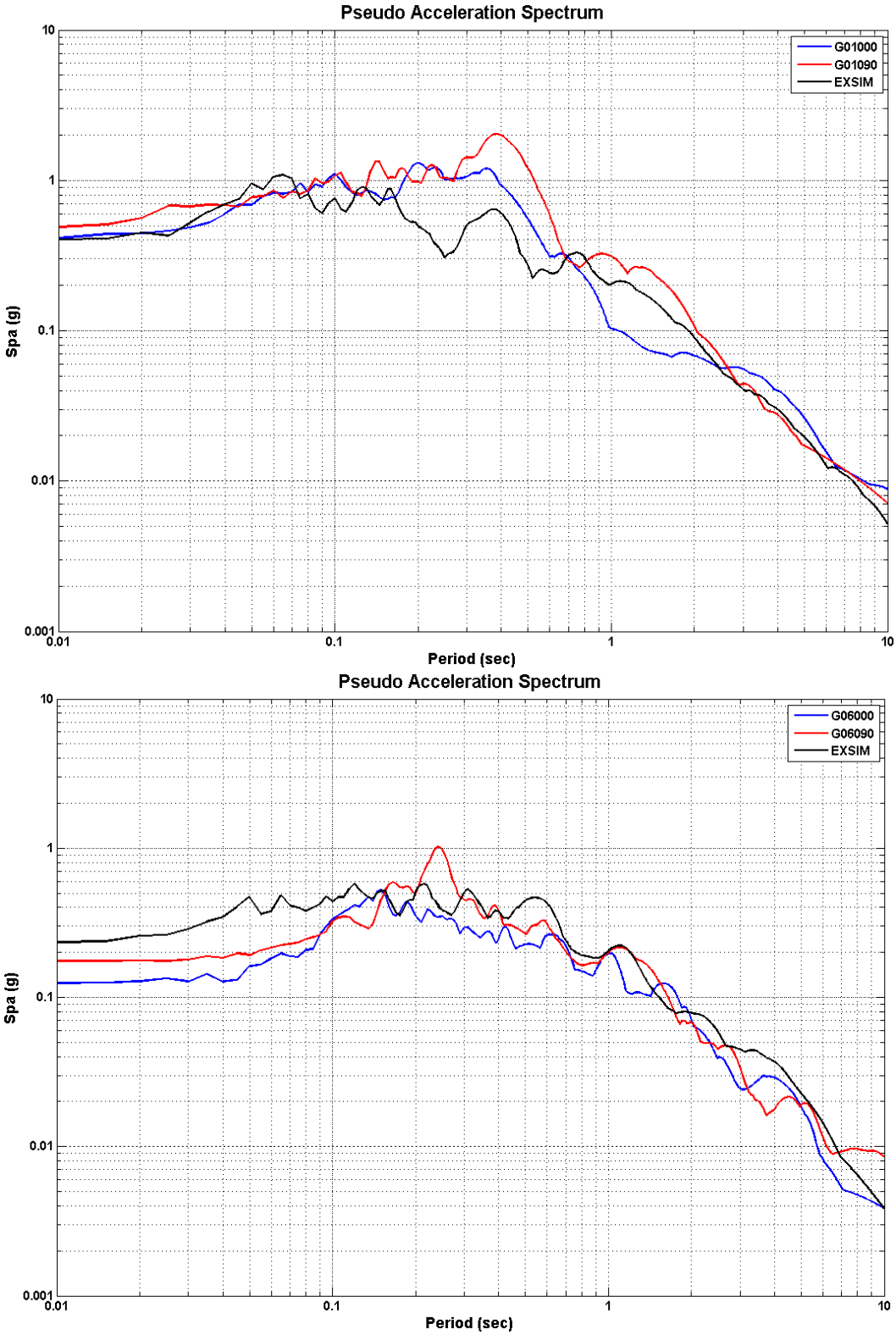


Figure 6.5 Pseudo acceleration spectrums for real and EXSIM simulated motions at Gilroy Array #1 (G01) and Gilroy Array #6 (G06) stations

6.3 Analysis of Lower Schultheis Road East Landslide

The Lower Schultheis Road East (LSRE) landslide is approximately 12 km southeast of the city of Los Gatos in Santa Cruz County and is situated on a short ridge immediately south of Summit Ridge region. LSRE landslide is one of the 20 deep-seated coherent landslides investigated by Keefer et al. (1998) and it is situated 13 km northwest of the 1989 Loma Prieta earthquake epicenter. The main scarp is 300 m long, 120 m wide and 4.5 ha in area. The average slope through the landslide profile is 20°. Measurements along the scarp showed 69 cm displacement (Keefer et al., 1998). The extent of the LSRE landslide as well as the surrounding landscape can be seen in Figure 6.6. 3D rendering of the landslide area is shown in Figure 6.7. The slope profile calculated from NED 1/3 arc-sec DEM is shown in Figure 6.8.

The area within and adjacent to the LSRE landslide contains several features that suggests preexisting landslides. This area was mapped by Cooper-Clark and Associates (1975) (Roberts and Baron, 1998) as a preexisting landslide deposit (see Figure 6.6). The geology consists of the Rices Mudstone Member of the San Lorenzo Formation (Tsr).

Cole et al. (1998) studied the Lower Schultheis Road West (LSRW) landslide which is approximately 500 m northwest of the LSRE landslide. Being in close proximity to each other, LSRW landslide geology is very similar to LSRE landslide. Cole et al. (1998) initially determined the shear strengths of samples collected from subsurface rupture zones by laboratory measurements. Subsequently, they performed backcalculations for static and dynamic loading conditions to improve the accuracy of calculated cumulative displacements. After the laboratory tests, they determined the best-fit values for the strength parameters as $\phi' = 24°$ and $c' = 22$ kPa. They reported a unit weight of 1920 kg/m^3 for the landslide material. Considering the similarity of the landslide materials, the same strength and unit weight values were used as a starting point for the dynamic analysis of LSRE landslide using FLAC.

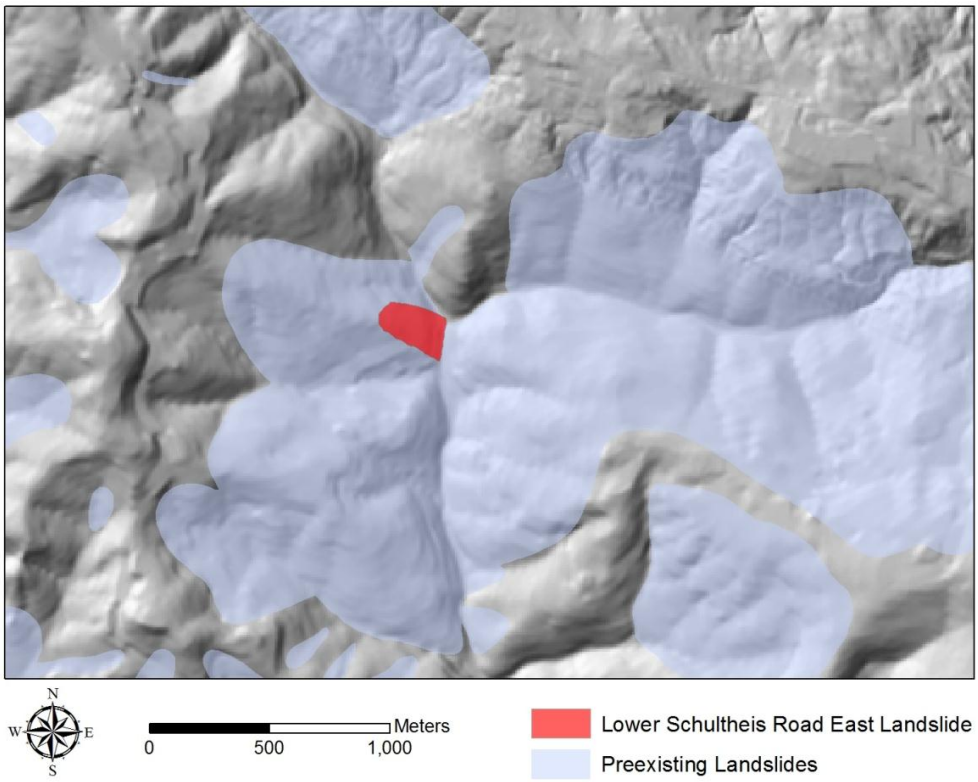


Figure 6.6 Hillshade of the Lower Schultheis Road East landslide

Figure 6.7 3D rendering of the Lower Schultheis Road East landslide

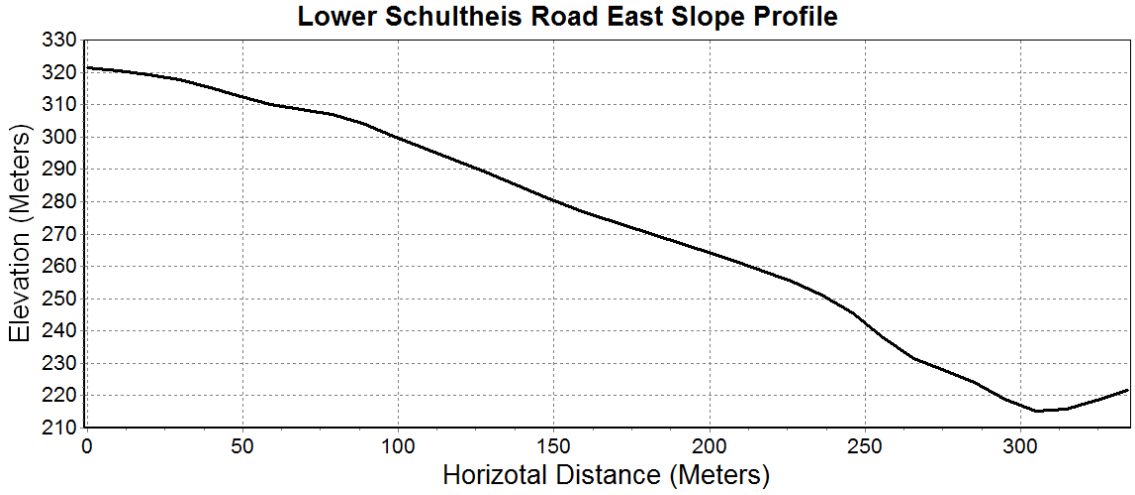


Figure 6.8 Slope profile of Lower Schultheis Road East Landslide calculated from NED 1/3 arc-second

6.3.1 FLAC model of Lower Schultheis Road East Landslide

Considering the dry ground conditions at the time of the Loma Prieta earthquake, the LSRE landslide is modeled as a homogenous dry slope. The slope height (H) is taken as 110 m and the slope angle (β) is 20°. A 50 m thick foundation layer is modeled under the base of the slope. In order to minimize the wave reflections, the lateral boundaries of the grid were modeled as free-field boundary conditions using viscous dashpots. A rigid boundary condition is assumed at the bottom of the foundation layer and the earthquake loading is applied at the rigid boundary as an acceleration time history. The element sizes are adjusted to provide accurate wave propagation. The material constitutive model is assumed elastic for the initial gravity stress analysis and for the dynamic analysis Mohr-Coulomb plasticity model is used. In order to avoid problems of excessive dilation during dynamic analysis, a non-associative flow rule (dilation angle, $\psi = 0°$) is used (Strenk, 2010). A full Rayleigh damping scheme (stiffness and mass-proportional components) is used to model the material damping. Rayleigh damping ($\xi = 2.5\%$) is applied at the mean frequency of the input ground motion. In dynamic analyses that use Mohr-Coulomb plasticity constitutive models, a considerable amount of energy dissipation occurs during

the plastic flow, therefore the use of a higher damping was deemed unnecessary (Itasca, 2008; Strenk, 2010).

The acceleration time history required for FLAC input is created using the finite-fault stochastic model (EXSIM) for Loma Prieta earthquake. Since the EXSIM model is already validated as presented in the previous section, the same input parameters in Table 6.2 are used for creating the artificial earthquake record at the site of LSRE landslide. Generic rock (Boore and Joyner, 1997) amplification coefficients are used in EXSIM simulation so that the generated artificial earthquake record can be applied directly at the rigid base of the FLAC model without any deconvolution analysis. The only required input for the EXSIM model is the latitude/longitude of the landslide location (37.119° N, 121.963° W). The generated artificial earthquake record and the corresponding 5% damped response spectra can be seen in Figures 6.9 and 6.10.

For the initial FLAC run, the strength parameters are assigned as $\phi' = 24°$ and $c' = 22$ kPa. A unit weight of 1920 kg/m^3 is assigned for the landslide material. The Poisson ratio of the slope material is assumed to be $\nu = 0.35$. There is no site-specific information for the shear wave velocity of the LSRE landslide site. However, information is available in the literature on the average shear wave velocities of different geologic units in California (Wills et al., 2000; Holzer et al., 2005; Wills and Clahan, 2006). Wills et al. (2000) defined San Lorenzo Formation (Ts) which also includes the Rices Mudstone Member (Tsr) as belonging to a C map category and assigned a mean shear wave velocity of 464 m/sec for this formation. Therefore an average shear wave velocity of 464 m/sec is used for the landslide material. Using the FLAC strength reduction method, the static factor of safety of the slope for the initial strength parameters is calculated as 1.44. Following the initial gravity stress analysis, a dynamic analysis is performed using the earthquake record created using EXSIM. FLAC Model Grid for LSRE landslide can be seen in Figure 6.11.

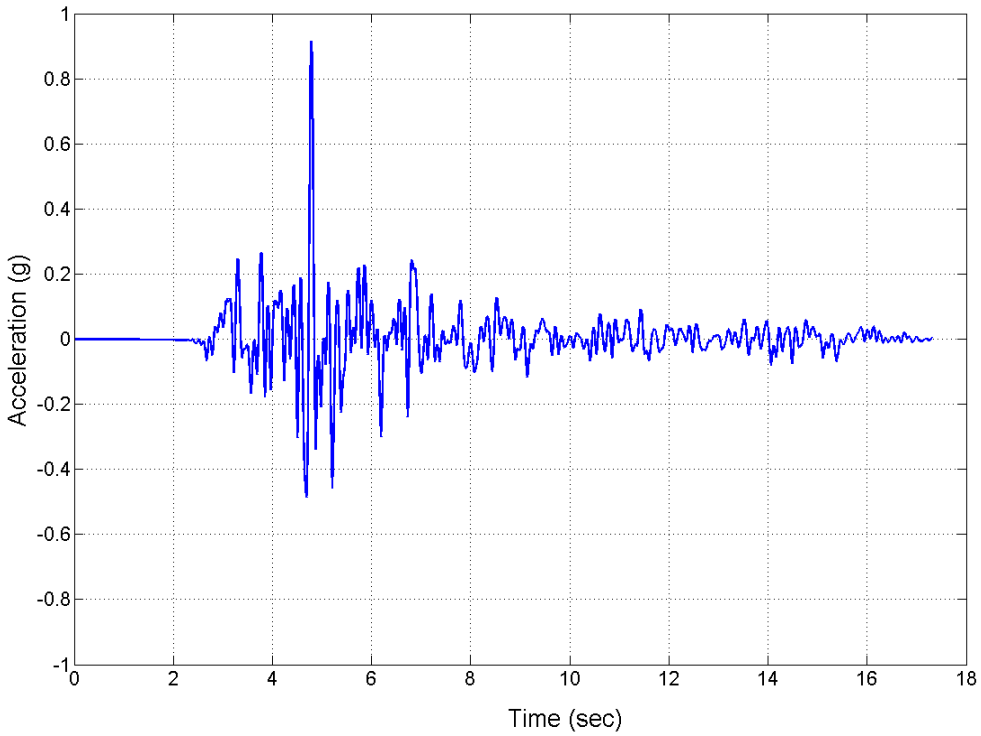


Figure 6.9 Earthquake record simulated by EXSIM for the Lower Schultheis Road East landslide

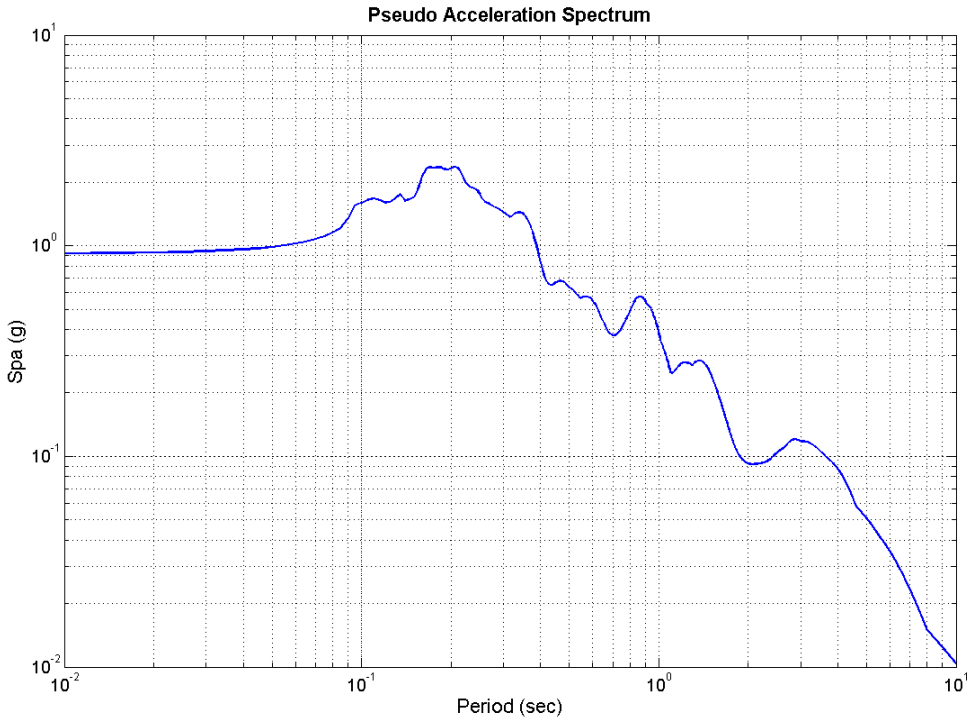


Figure 6.10 Response spectra of the earthquake record simulated by EXSIM for the Lower Schultheis Road East landslide

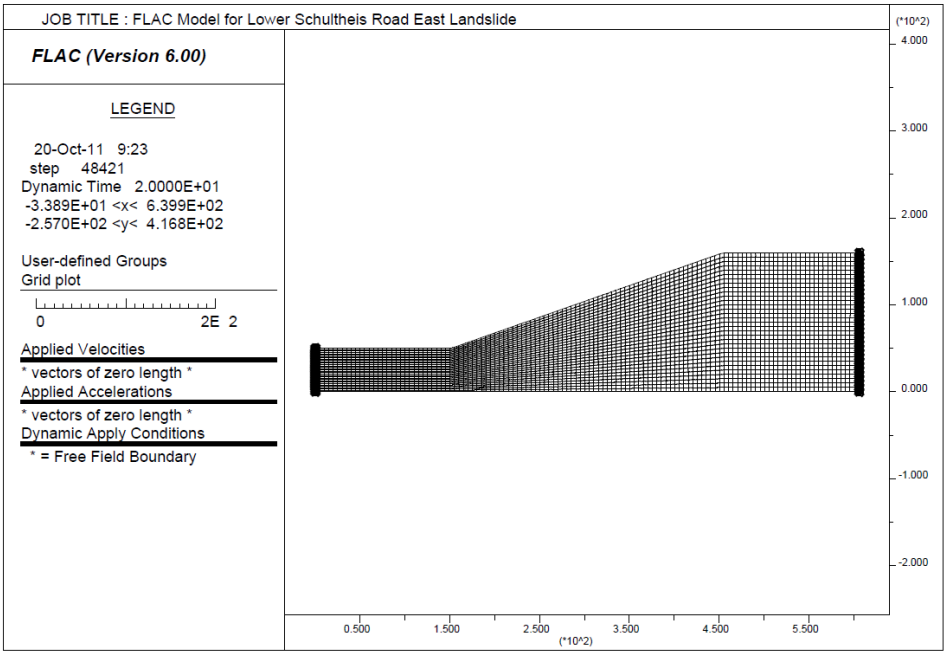


Figure 6.11 FLAC model grid for the Lower Schultheis Road East landslide

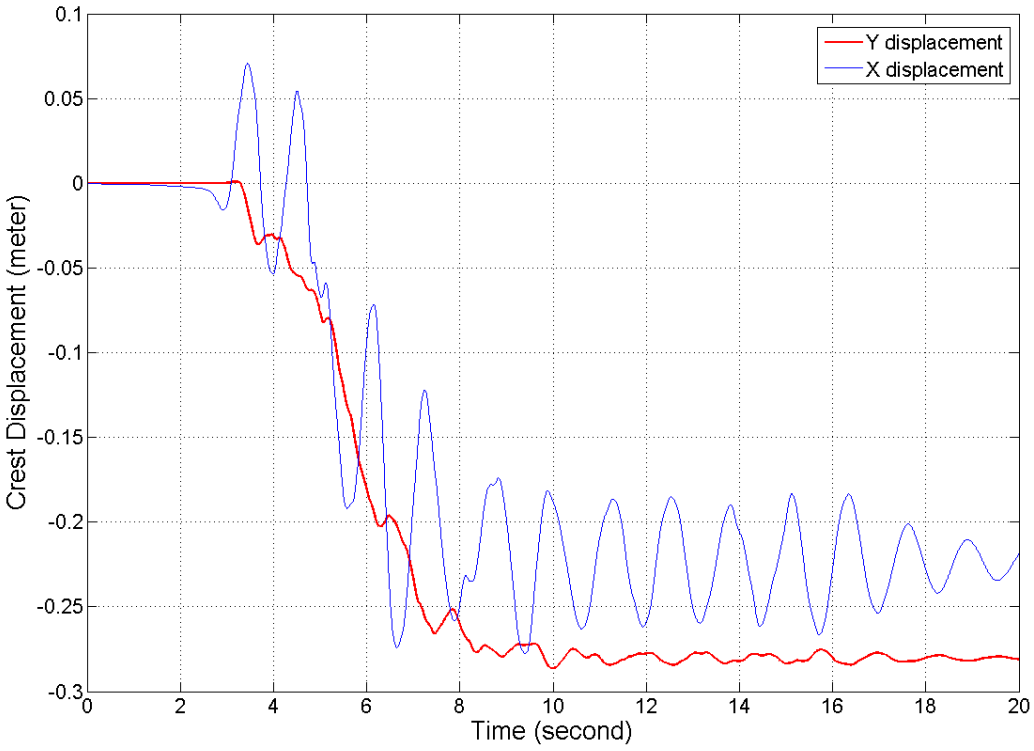


Figure 6.12 Crest displacements calculated with FLAC for the LSRE landslide

For the initial FLAC run the x and y components of crest displacements versus time can be seen in Figure 6.12. The total crest displacement at the end of the earthquake loading is calculated to be 36 cm. However, the measurements along the scarp of the LSRE landslide showed 69 cm total displacement. Cole et al. (1998) stated that laboratory strengths may be too high to accurately predict displacements and they backcalculated the strength parameters using the observed performance of the slope. A similar backcalculation methodology is used for the LSRE landslide. A parametric study is performed by changing the effective friction angle ϕ' while keeping all other parameters including the earthquake loading constant. The change of calculated crest displacements with respect to the assigned effective friction angles is shown in Table 6.4. As it can be seen in Table 6.4, 65 cm displacement is calculated with $\phi' = 21°$, which is very close to the measured displacement 69 cm. Backanalysis of the shear strength of the landslide material for static stability also can help define a realistic range of friction angles (Keefer, 1991; Cole et al., 1998). For example in the case of the LSRE landslide, it is known that the present geometry of the landslide is stable, therefore the static factor of safety should be larger than 1. This information provides a minimum limit for the shear strength of the landslide material. As it can be seen in Table 6.4 if the effective friction angle is smaller than $17°$, the slope becomes statically unstable.

In the same parametric study, the performances of the ANN regressor created in Chapter 3 for estimating the static factor of safety and the SVM classifier created in Chapter 4 for predicting the displacement classes of homogenous dry slopes are also investigated. For each different effective friction angle value, static F.S. is calculated using FLAC's strength reduction method. The predicted factor of safety column in Table 6.4 is calculated using the ANN regressor created and trained in Chapter 3. The ANN regressor performed very well and predicted the static F.S. values of LSRE landslide very close to real calculated values.

Table 6.4 Parametric study showing the crest displacements of LSRE landslide model with respect to effective friction angle and the prediction performance of metamodels

ϕ' (^{o})	Calculated F.S.	Predicted F.S.	Calculated Crest Displacement (cm)	Calculated Crest Displacement Class	Predicted Crest Displacement Class
16	0.97	0.95	276	High	High
17	1.04	1.04	187	High	High
18	1.09	1.07	134	High	High
19	1.15	1.14	104	High	High
20	1.21	1.22	83	High	High
21	1.27	1.27	65	High	High
22	1.33	1.35	51	High	Medium
23	1.38	1.40	42	Medium	Medium
24	1.44	1.43	36	Medium	Medium
25	1.50	1.50	30	Medium	Medium
26	1.57	1.58	25	Medium	Medium
27	1.63	1.63	20	Medium	Medium
28	1.69	1.67	16	Medium	Medium
29	1.75	1.74	12	Medium	Medium
30	1.81	1.82	10	Medium	Medium
31	1.88	1.89	8	Medium	Medium
32	1.94	1.94	7	Medium	Medium
33	2.01	2.02	6	Medium	Low
34	2.08	2.08	5	Medium	Low
35	2.15	2.15	4	Low	Low

Crest Displacement ≤ 5 cm **Low**

5 cm < Crest Displacement ≤ 50 cm **Medium**

50 cm < Crest Displacement **High**

The calculated crest displacements are assigned hazard classes according to Table 4.6. Subsequently, SVM classifier created in Chapter 4 is used to predict the hazard classes using the input parameters including input earthquake intensity measures (IM), slope material and geometry. The SVM classifier performed very well except at the class boundaries 5 cm and 50 cm.

The depth to bedrock is assumed as 50 m in the analysis. The effect of depth to bedrock is also investigated with $\phi' = 21°$ and all the other parameters kept constant. The depth to bedrock is changed from 10 m to 100 m. The change in crest displacement with respect to bedrock can be seen in Figure 6.13. Small fluctuations are expected because changing the depth to bedrock essentially changes the 1D period of the slope. However, the change in crest displacement is negligible.

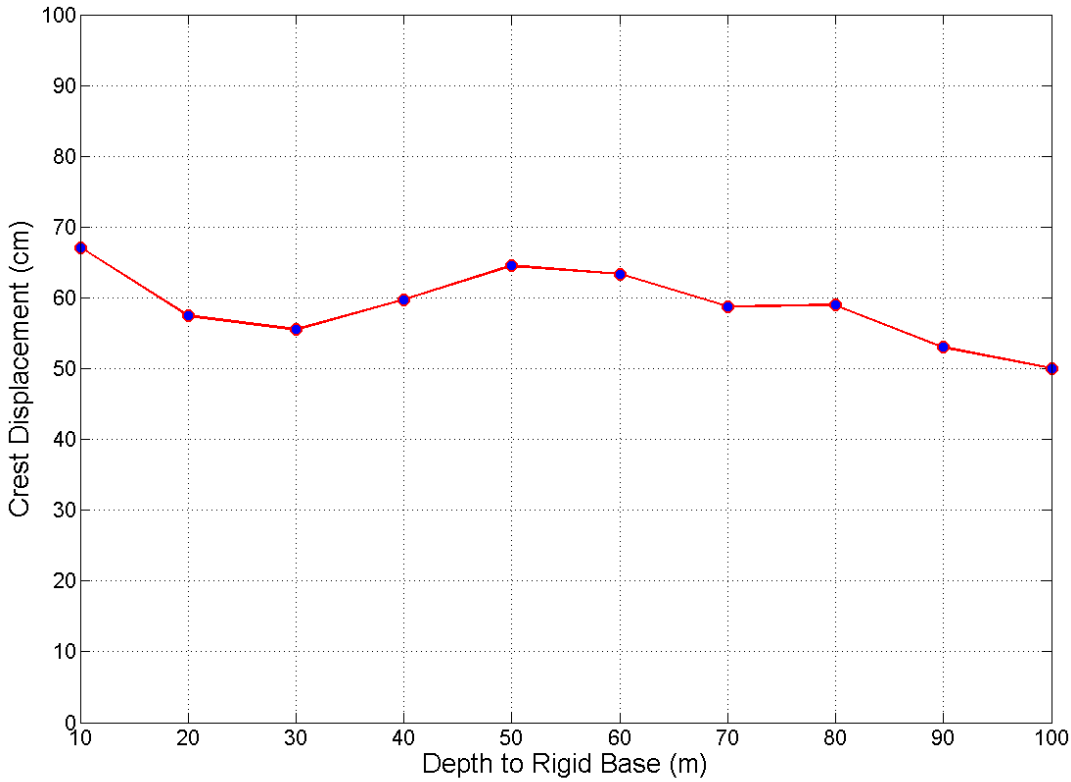


Figure 6.13 Crest displacements calculated with respect to depth to rigid base

6.4 Pilot Study

The area affected by the 1989 Loma Prieta earthquake extends inland from Pacific cost, through the broad lowlands around San Francisco Bay and Monterey Bay, to the parts of the Coast Ranges of Central California (Keefer, 1998). Topography in the earthquake region ranges from the near vertical cliffs bordering the Pacific Ocean and the steep slopes of California Coast Ranges to flat bay margins, valley floors and coastal terraces (Keefer, 2000). Most of the landslides occurred in the southern Santa Cruz Mountains which includes the Laurel quadrangle through which the surface projection of the fault rupture plane passes (see Figure 6.14). Topography in the southern Santa Cruz Mountains ranges from steep rugged ridges separated by narrow canyons to gently rolling hills. Keefer and Manson (1998) classified 74% of the landslides in this region as disrupted slides and falls and 26% of the landslides as coherent landslides. The coherent landslides were rotational slumps and translational block slides which had volumes ranging up to tens of millions of cubic meters (Keefer, 1991; Keefer and Manson, 1998).

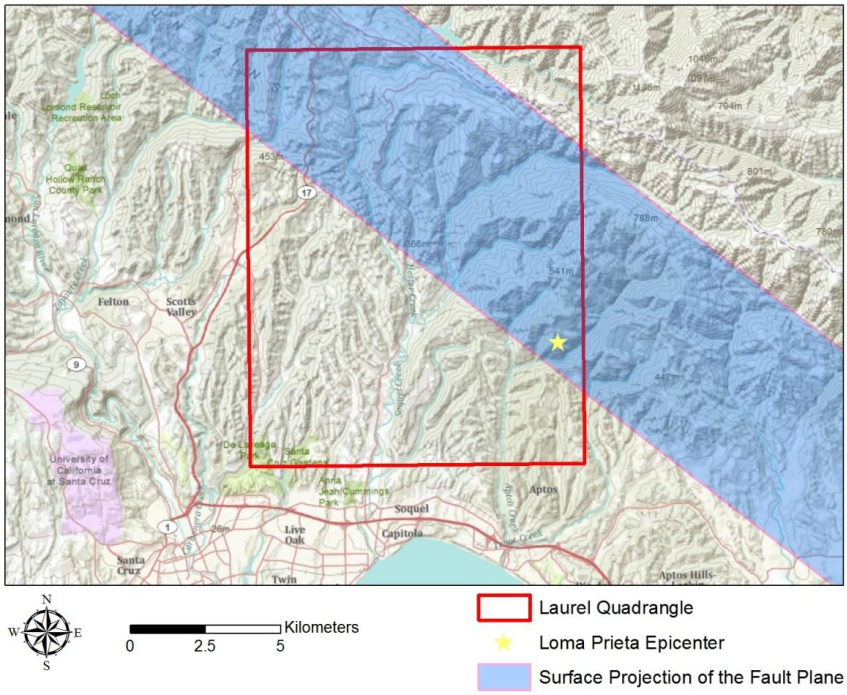


Figure 6.14 Location of Laurel quadrangle

In the southern Santa Cruz Mountains southwest of the San Andreas Fault, predominant bedrock units are Tertiary sedimentary rocks and smaller bodies of intrusive, volcanic, and metamorphic rocks (McLaughlin et al., 2002). A geologic map of Laurel quadrangle can be seen in Figure 6.15. The names and the total areas of the geologic units mapped in Laurel quadrangle can be seen in Table 6.5. The most spread sedimentary units are the Purisima Formation (Tp), Vaqueros Formation (Tv), Butano Sandstone (Tbc, Tbm, Tbs, and Tbu), San Lorenzo Formation (Tsr and Tst), Lambert Shale (Tla), Monterey Formation (Tm), Santa Cruz Mudstone (Tsc) and Lompico Sandstone (Tlo). These sedimentary units are poorly to moderately consolidated, varyingly weathered, intensely folded, and locally sheared and faulted (Keefer, 2000).

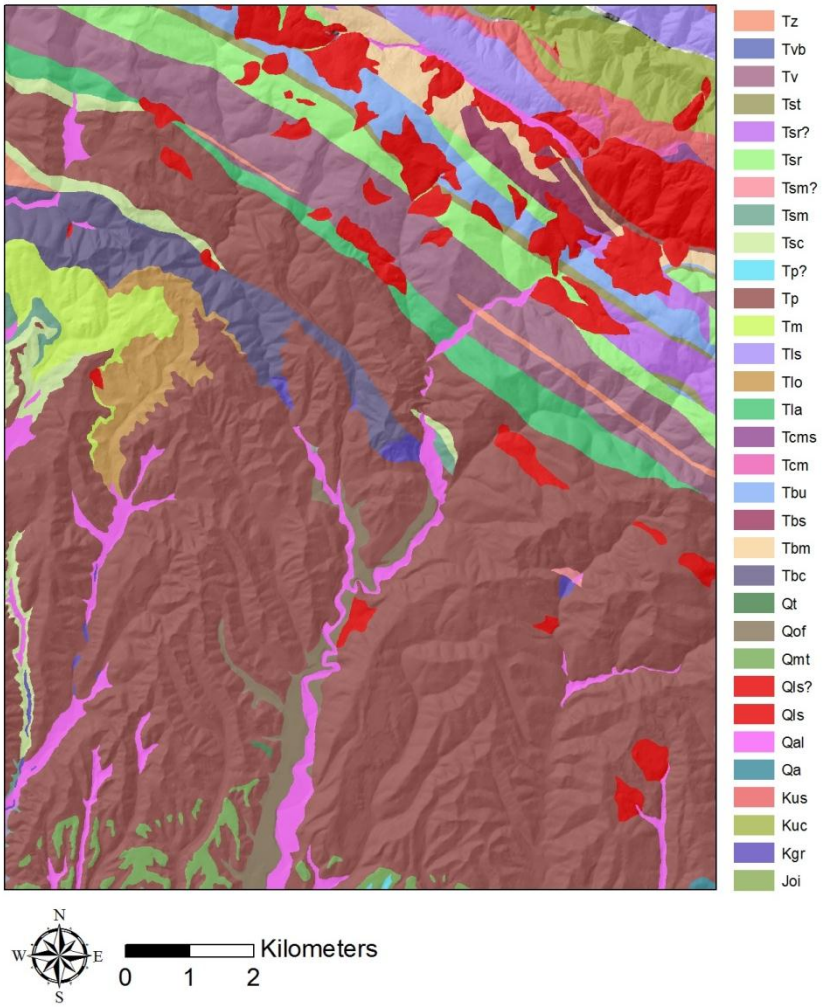


Figure 6.15 Geologic map of Laurel quadrangle (McLaughlin et al., 2002)

Preexisting landslide deposits are widespread in Laurel quadrangle. Keefer et al. (1998) reported that most of the coherent deep-seated landslides in the southern Santa Cruz Mountains exhibited geomorphic indications of previous landslide movement. The preexisting landslide deposits are shown in the geologic map prepared by McLaughlin et al. (2002) with the symbol Qls (see Figure 6.15). Preexisting landslide deposits in Laurel quadrangle are also mapped by Cooper-Clark and Associates (1975) (Roberts and Baron, 1998). The various degrees of certainty in identification are reflected by the classification of preexisting landslide deposits such as “definite”, “probable” or “questionable” (see Figure 6.16).

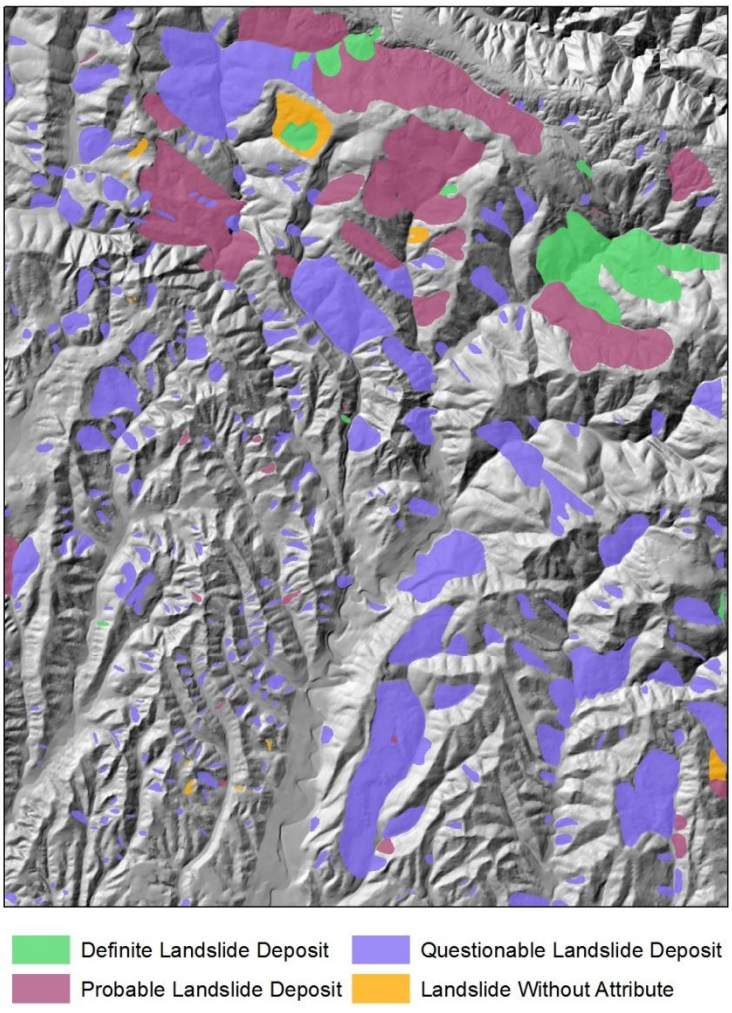


Figure 6.16 Preexisting landslide deposits in Laurel quadrangle (Roberts and Baron, 1998)

Table 6.5 Geologic units, areas in Laurel quadrangle, shear strengths for geologic units (after (McCrink and Real, 1996))

Unit	Name	Area (km^2)	Mean Effective Friction Angle	Group Median Cohesion (kPa)	Number of Shear Strength Tests	Average Shear Wave Velocity (m/sec) 1
Tp	Purisima Formation	82.29	33.1	25.2	79	372
Tv	Vaqueros Formation	11.04	33.3	24.3	22	464
Tbc	Butano Sandstone (Conglomerate)	5.11	28.9	26.2	26	515
Tsr	Rices Mudstone Member of San Lorenzo Formation	5.01	34.9	27.0	35	464
Tla	Lambert Shale	3.89	12.8	35.9	6	576
Kuc	Great Valley Sequence (Conglomerate)	2.96	30*	55*	n.d.	724
Tm	Monterey Formation	2.87	29.5	25.4	2	464
Tsc	Santa Cruz Mudstone	2.75	29.5	25.4	0#	464
Tbu	Butano Sandstone (undivided)	2.69	28.9	26.2	26	515
Tlo	Lompico Sandstone	2.59	36.0	23.9	5	464
Tls	Loma Chiquita Ridge Sandstone and Mudstone	2.59	30*	55*	n.d.	464
Tbm	Butano Sandstone (Mudstone)	1.89	28.9	26.2	26	515
Kus	Great Valley Sequence (Sandstone and Shale)	1.66	32.5	62.5	n.d.	724
Tbs	Butano Sandstone (Sandstone)	1.10	28.9	26.2	26	515
Tst	Twobar Shale Member of San Lorenzo Formation	1.03	27.4	25.4	5	464

1 Average Shear Wave Velocities adopted from (Wills et al., 2000; Wills and Clahan, 2006), n.d. = not determined, *=adopted from (Miles and Keefer, 2009)

McCrink and Real (1996) undertook an extensive compilation of shear strength data for geologic units in the Laurel quadrangle (see Table 6.5). This compilation of data

obtained from geotechnical boring-logs, laboratory direct-shear test results, and geologic site investigations (Keefer, 2000). McCrink and Real (1996) concluded that mean values of $\phi'$and median values of c' for each formation or group of formations were most representative and these values are shown in Table 6.5. Keefer (2000) in his analysis of Loma Prieta landslides assumed dry conditions and a uniform dry unit weight of 2000kg/m^3 for all geological units. Average shear wave velocities of different geologic units in Table 6.5 are obtained from the literature (Wills et al., 2000; Wills and Clahan, 2006). The assumption of homogenous soils and assignment of average material properties to geological units are common for regional scale modeling of earthquake induced landslides (Luzi and Pergalani, 1996; McCrink and Real, 1996; Mankelow, 1997; Miles and Ho, 1999; Jibson et al., 2000; Refice and Capolongo, 2002; Miles and Keefer, 2009). Therefore, the material shear strength values and shear wave velocities in Table 6.5 will initially be assigned to respective geological units for this study. Considering the dry conditions of the Loma Prieta landslides a uniform dry unit weight of 2000 kg/m^3 will be assigned to all geological units (Keefer, 2000).

There are several reports in the literature for the shear strength of the preexisting landslide deposits. A list of shear strength values assigned to preexisting landslide materials by different researchers can be seen in Table 6.6. The shear strength values reported by Coles et al. (1998) and Keefer (1991) are for site specific deep-seated coherent landslides reactivated during Loma Prieta earthquake. The shear strength values reported by Miles and Keefer (2009) are used in a regional earthquake triggered landslide hazard analysis conducted for Laurel quadrangle. Jibson et al. (2000) assigned the values listed in Table 6.6 for a regional hazard analysis studying the landslides triggered by 1994 Northridge earthquake. Wilson et al. (2008) used the shear strength values reported for a seismically induced landslide hazard analysis of moderate to deep-seated landslides for M 7.8 Southern San Andreas earthquake scenario (ShakeOut). The most important reason for this variability in shear strength values stems from the fact that composition of

the preexisting landslide deposit depends on the source material. Considering the values given by Miles and Keefer (2009) represent general values for the preexisting landslide deposits in Laurel quadrangle and also considering the observation that the values backcalculated from the LSRE landslide in this study are closer to those reported in Miles and Keefer (2009), these values are initially assigned to preexisting landslide deposits in this study.

Table 6.6 Shear strength values assigned for preexisting landslide deposits

Area	Effective Friction Angle	Effective Cohesion (kPa)	Reference
Laurel Quadrangle	20°	24	(Miles and Keefer, 2009)
Laurel Quadrangle	24°	22	(Cole et al., 1998)
Laurel Quadrangle	25°	21	(Keefer, 1991)
South California	30°	24	(Jibson et al., 2000)
South California	25°	33.5	(Wilson et al., 2008)

The National Elevation Dataset (NED) 1/3 arc-second digital elevation model (DEM) assembled by the U.S. Geological Survey (USGS) is used for the delineation of slope profiles in Laurel quadrangle. The horizontal resolution of the dataset is approximately 10 m and the vertical accuracy is +/- 7m. DEM for Laurel quadrangle and the hillshade of the region can be seen in Figure 6.17. The lowest elevation in the quadrangle is 13 meters and highest elevation is 831 meters.

First, watersheds in the Laurel quadrangle are delineated using the steps explained in the flowchart in Figure 5.1. The delineated watersheds and drainage lines can be seen in Figure 6.18. Flat areas which have slope values smaller than 10^{o} are deleted from watershed polygons.

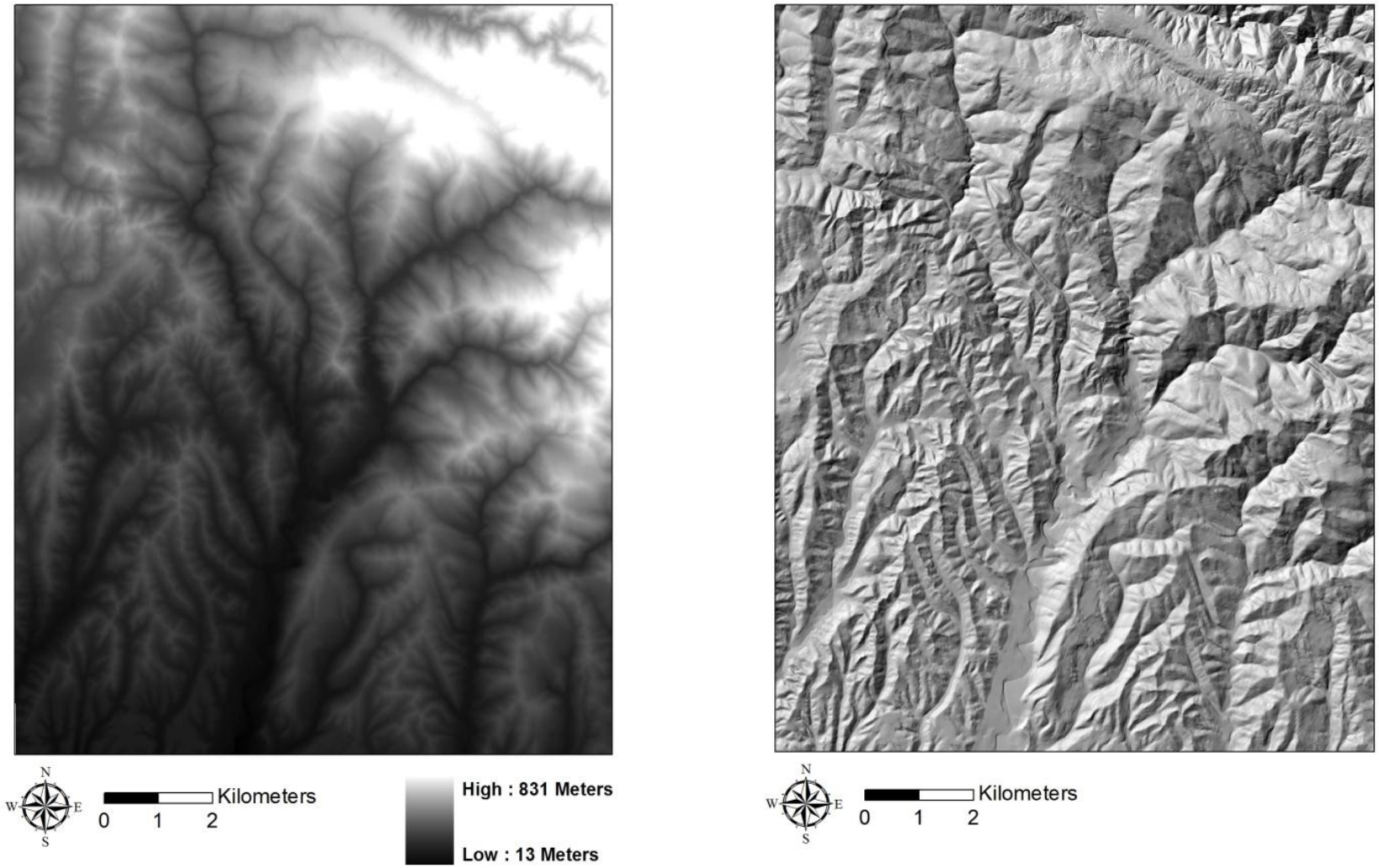


Figure 6.17 NED 1/3 Arc Second digital elevation model (DEM) (left) and hillshade of Laurel quadrangle (right)

A buffer analysis is performed with a negative buffer distance (1 meter) on normal watershed polygons to avoid problems at the boundary of two neighboring watersheds. A code written in Visual Basic is used to create points in 50 m intervals along the ridge lines. The points created along the ridge lines can be seen in Figure 6.18. Steepest descent approach explained in section 5.2.2 is used to create slope profile polylines. A Visual Basic code is used to draw steepest descent polylines from each point until they intersect a drainage line or a flat area. The steepest descent polylines drawn from the points created along the ridge lines can be seen in Figure 6.19. These polylines may or may not be straight lines depending on the topography. A straight slope profile polyline is drawn from each ridge point to the point where the steepest descent polyline drawn from that point intersects a drainage line or a flat area. Slope profile polylines can be seen in Figure 6.19. Next, the slope profile polylines are converted to three dimensional polylines so that the geometric properties of the profiles such as slope angle, height can be determined. All of the slope profiles are assumed to be linear and the attribute tables of the slope profile polylines are populated with the calculated height and slope angle values for each profile. Slope profiles with slope angles less than 10° are deleted from the database. A total number of 6104 slope profiles are delineated. The slope angle of these profiles ranges from 10° to 44°. The height of the profiles ranges from 10m to 350 m.

Next, using the spatial join tool in ArcGIS slope profiles are assigned to their corresponding geologic units shown in Figure 6.15. However, slope profiles intersecting the preexisting landslide deposits in Figure 6.16 (including all classes) were assigned the shear strength parameters of preexisting landslides irrespective of their original geologic unit. If a slope profile was intersecting more than one geologic unit, the unit which contains a longer part of the polyline is assigned to that profile. The slope profile attribute table is populated with the corresponding shear strength and shear wave velocity values.

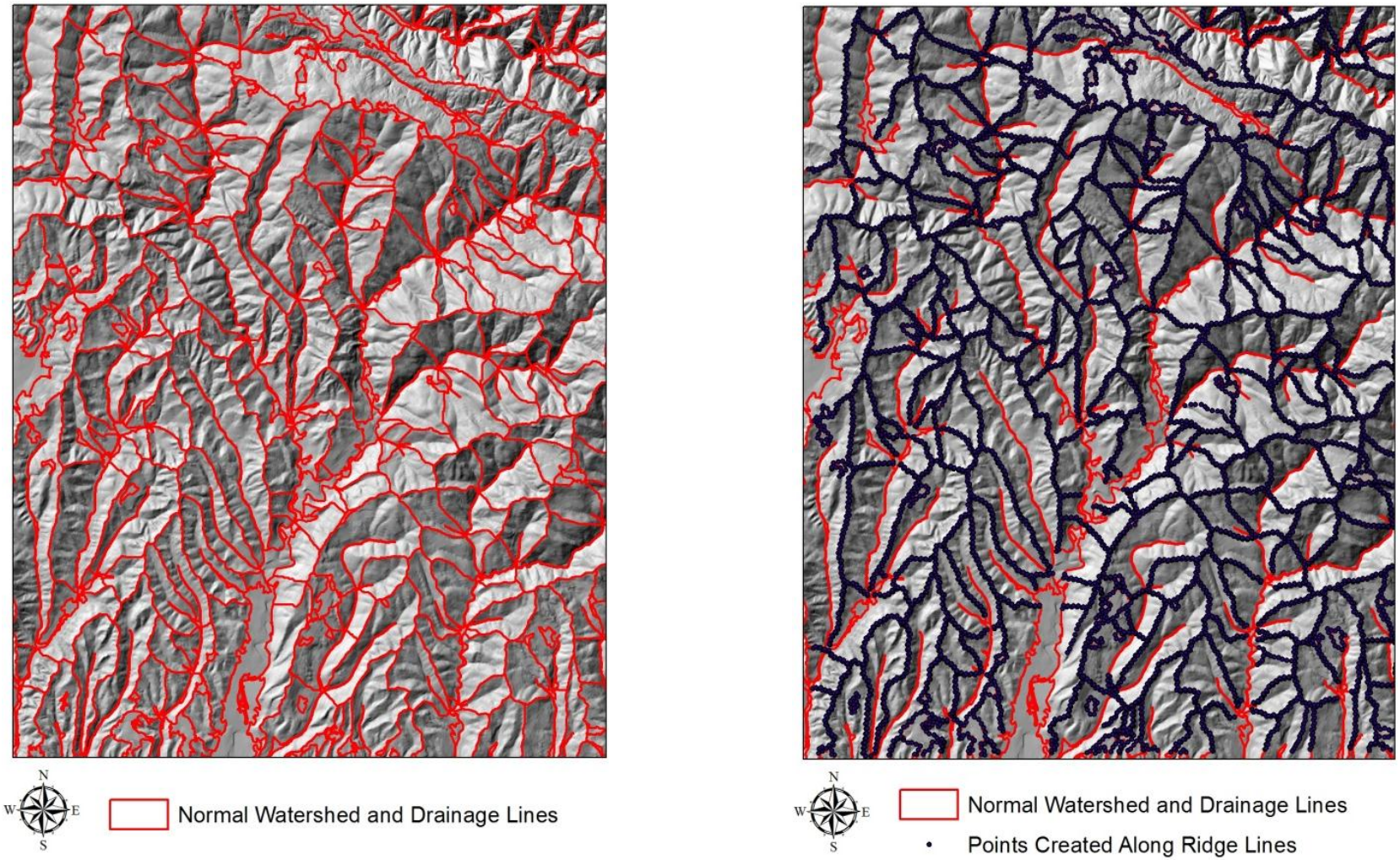


Figure 6.18 Normal watersheds and drainage lines (left) and points created along ridge lines (right)

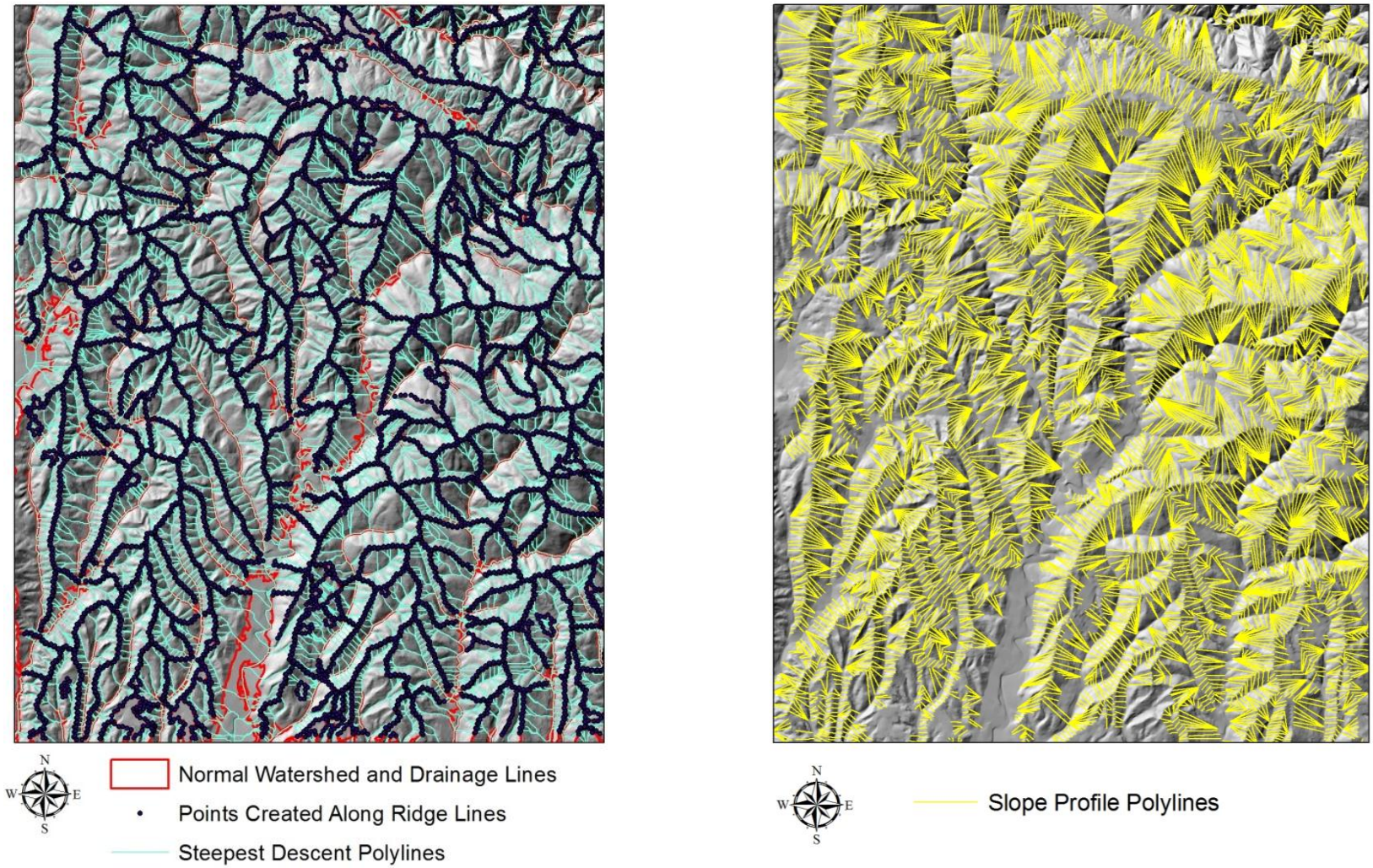


Figure 6.19 Steepest descent polylines (left) and slope profile polylines (right)

Considering the dry conditions during the Loma Prieta earthquake, it is assumed that the ground water table was below the failure surface for all slope profiles. Using the ANN regressor created and trained in Chapter 3, the static factor of safety values of the slope profiles are calculated. The color coded static factor of safety of the slope profiles can be seen in Figure 6.20. All of the slope profiles have static factor of safety values higher than 1. Therefore, the shear strength values assigned for the geologic units satisfy the lower limits for static stability.

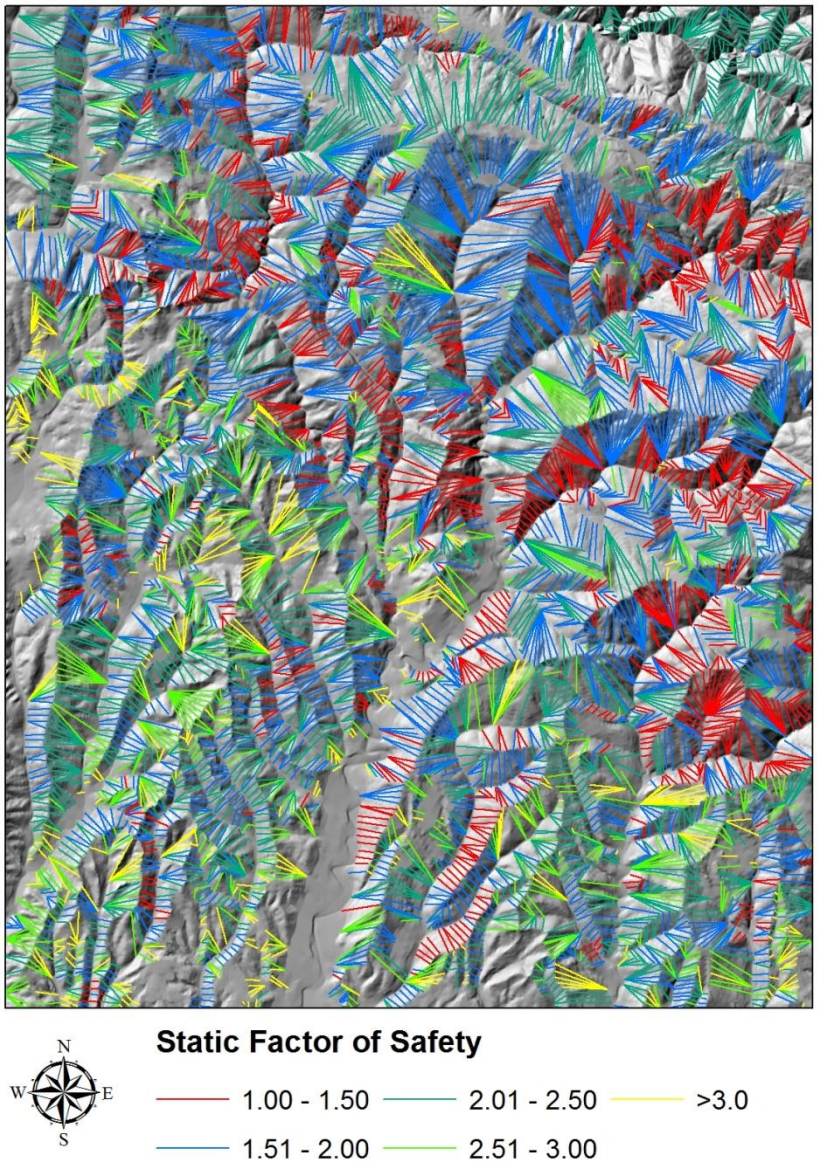


Figure 6.20 Color coded static factor of safety of slope profiles

The artificial earthquake records simulating the 1989 Loma Prieta earthquake are created using the EXSIM finite-fault model with the input model parameters shown in Table 6.2. All the records are generated for generic rock conditions (Boore and Joyner, 1997). Instead of generating separate records for each slope profile polyline, records are created using the latitude and longitude of each watershed center and assigned to the polylines inside the watershed. For each of the record generated, intensity measures (IM) are calculated and the attribute table of the slope profiles is populated with the IM values calculated for each profile. Using the SVM classifier created and trained in Chapter 4 with a combination of artificial and real earthquake records, each slope profile is assigned one of the three hazard classes "Low", "Medium", and "High". Color coded predicted hazard classes and coherent landslides triggered by Loma Prieta earthquake can be seen in Figure 6.21.

Other than the 16 large deep-seated landslides reported by Keefer et al. (1998), only 44 coherent landslides in Laurel quadrangle have reported displacement values in the landslide inventory. A total of 305 coherent landslides are reported in Laurel quadrangle, these landslides are reported as point feature class. 55 of these landslides are on the flat areas (slope angle < 10°) that were not taken into consideration during delineation of slope profiles. Most of these 56 landslides' geomorphology is defined in the inventory as streambank or swale. The predicted hazard classes and coherent landslide inventory have a high correlation. From the remaining 250 coherent landslides almost 90% have a slope profile in a 50m radius which has a "Medium" or "High" hazard class. All of the profile polylines intersecting large coherent landslides in the Laurel quadrangle are predicted as high hazard classes. However, almost 30% of the slope profiles which have hazard classes "Medium" or "High" do not have a reported landslide in 50m buffer from the profile. This means that the strength values assigned for these slope profiles are lower than their actual values.

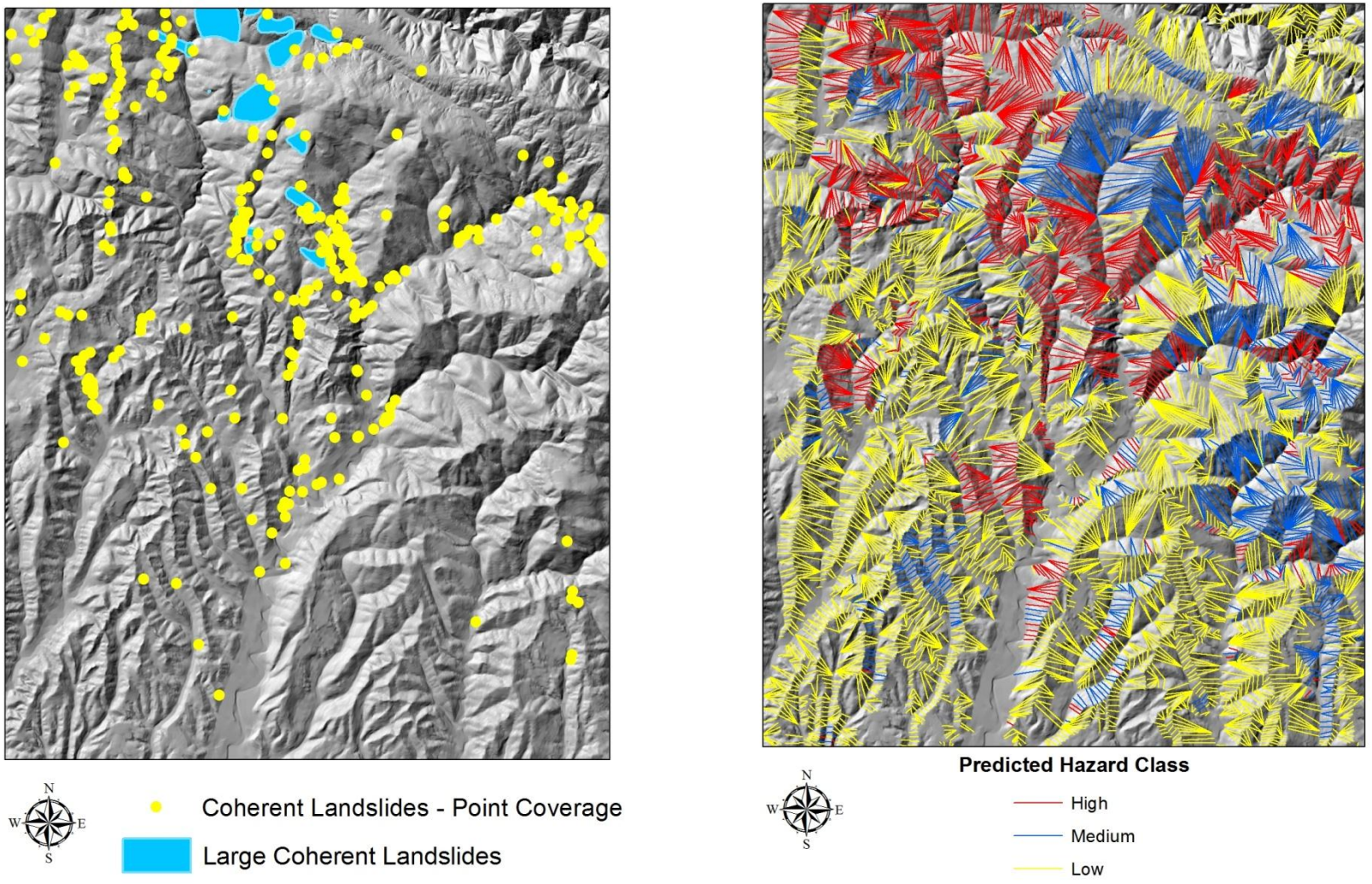


Figure 6.21 Loma Prieta landslide inventory (left) and predicted hazard classes for Loma Prieta earthquake

Using the SVM classifier, the strength values of these slope profiles are backcalculated using their performance during Loma Prieta earthquake. While keeping all other parameters the same, their effective friction angle values are increased until they have a "Low" hazard class under the same earthquake loading. The slope profiles with the new backcalculated shear strength values are used for the seismic stability analysis for the scenario earthquake.

The working group on California earthquake probabilities published a report on earthquake probabilities in the San Fransisco Bay region between 2002-2031 (WG, 2003). In this report they presented results that came from a comprehensive analysis led by USGS and involving input from a broad group of geologists, seismologists, and other earth scientists. They concluded that there is a 62% probability of a major, damaging earthquake striking the greater San Francisco Bay region between 2002 and 2031. One of the fault segments that they investigated is the part of the San Andreas Fault that passes through Laurel quadrangle named SAS (Santa Cruz Mountains) fault segment (see Figure 6.22). The working group defined fault segment as the building block for each fault, the shortest section considered capable of repeatedly rupturing to produce large earthquakes. SAS is a 62-km long segment that extends from Los Gatos (LG) to San Juan Bautista (SJB). WG (2003) reported the preferred length and width of the SAS fault segment as 62 km and 15 km respectively. The most recent rupture on this fault segment was 1906 San Francisco earthquake. The mean magnitute of the expected earthquake on SAS fault segment is reported to be 7.03 (WG, 2003). The mean probabilty of rupture on the SAS fault segment between years 2002-2031 is reported as 0.026.

This M_w = 7.03 scenario earthquake on SAS segment of the San Andreas Fault is modeled as a finite-fault model using EXSIM. It is modeled as a strike-slip fault. The EXSIM model parameters can be seen in Table 6.7. Similar to the Loma Prieta earthquake simulation, the artificial earthquake records are generated only for delineated watersheds and the calculated intensity measure (IM) values are assigned to the slope

profiles inside that watershed polygon. Dry condition is assumed for the soil profiles. Using the SVM classifier created and trained in Chapter 4, the hazard classes are predicted for each slope profile. The predicted hazard classes for the scenario earthquake can be seen in Figure 6.23. As can be seen in the Figure 6.23 most of the coherent landslides triggered by the Loma Prieta earthquake are likely to be reactivated for the scenario earthquake. Taking into consideration the longer fault segment ruptured, rupture plane being close to the surface and the slightly higher magnitude earthquake, there is a 20% increase in the slope profiles with hazard classes "High" and there is almost 30% increase in the slope profiles with "Medium" hazard classes compared to Loma Prieta earthquake.

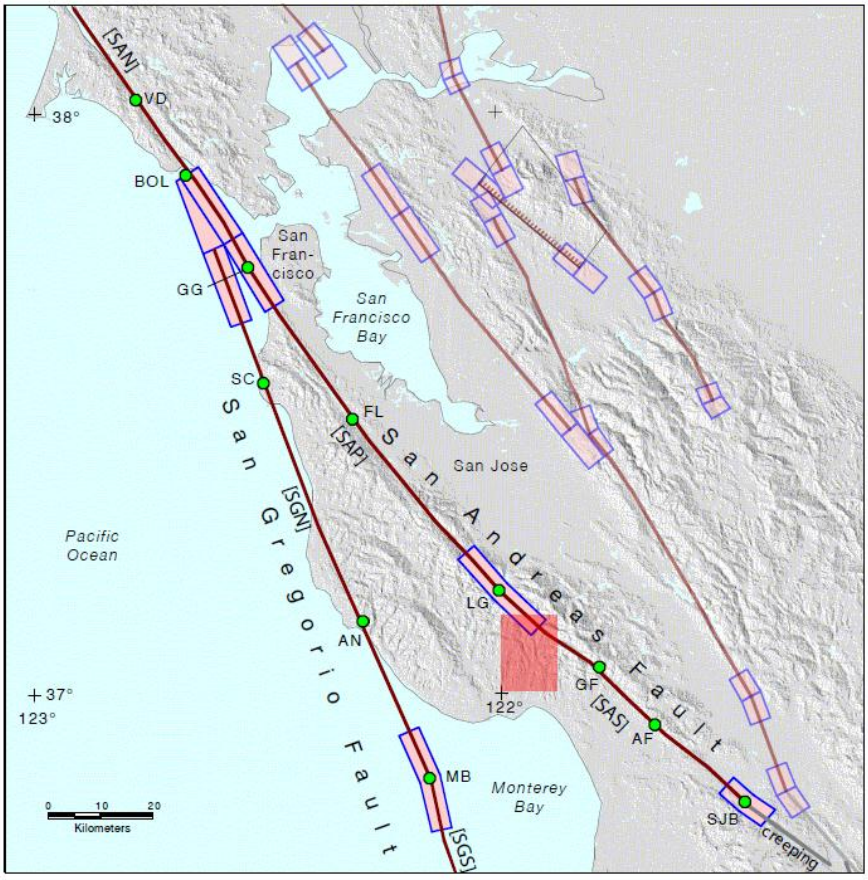


Figure 6.22 SAS segment of San Andreas Fault (from WG (2003)) (red rectangle is Laurel quadrangle)

Table 6.7 EXSIM modeling parameters for the scenario earthquake on SAS segment of San Andreas Fault (M_w=7.03)

- Fault Orientation: Strike 128^o , Dip 90^o
- Depth of top: 0 km
- Fault dimensions: Length 62km, Width 15 km
- Number of subfaults: 12 lengthwise, 4 widthwise
- Geometrical Attenuation:
  $R < 40km : R^{-1}$
  $40km \leq R : R^{-0.5}$
- $Q(f) : 187f^{0.56}$
- Crustal shear-wave velocity: 3.7 km/sec
- Crustal Density: 2.8 g/cm^3
- Stress drop (bars): 60
- Site amplification: generic rock (Boore and Joyner, 1997)
- Kappa: 0.030
- Pulsing area percentage: 25%
- Windowing function: Saragoni-Hart

6.5 Summary

An integrated framework is developed by combining the soft computing methods for predicting the performance of slopes under static and dynamic loading with an automated slope profile delineation method. The Lower Schultheis Road East landslide was analyzed in detail as a case history using dynamic FLAC analyses. Using the same case history, the performance of the SVM classifier was investigated by comparing the class calculated using FLAC analysis and the class predicted using SVM classifier. SVM classifier showed a very good performance predicting the hazard classes with given slope material and geometric properties, and intensity measures of the earthquake loading. The framework is evaluated using the landslide inventory of the 1989 Loma Prieta earthquake (Mw = 6.9) within the 7.5 min. Laurel, CA quadrangle. Given the average shear strength values for the geological units, the model over predicted the landslide occurrence. However, as a result of the flexibility and efficiency of the model, it is relatively easy to backcalculate and calibrate the material shear strength parameters assigned to the slope

profiles using their performance during a previous earthquake. After backcalculating the shear strength parameters for the slope profiles, a seismic landslide hazard analysis was performed for a future scenario earthquake (Mw = 7.03) on the SAS segment of the San Andreas Fault.

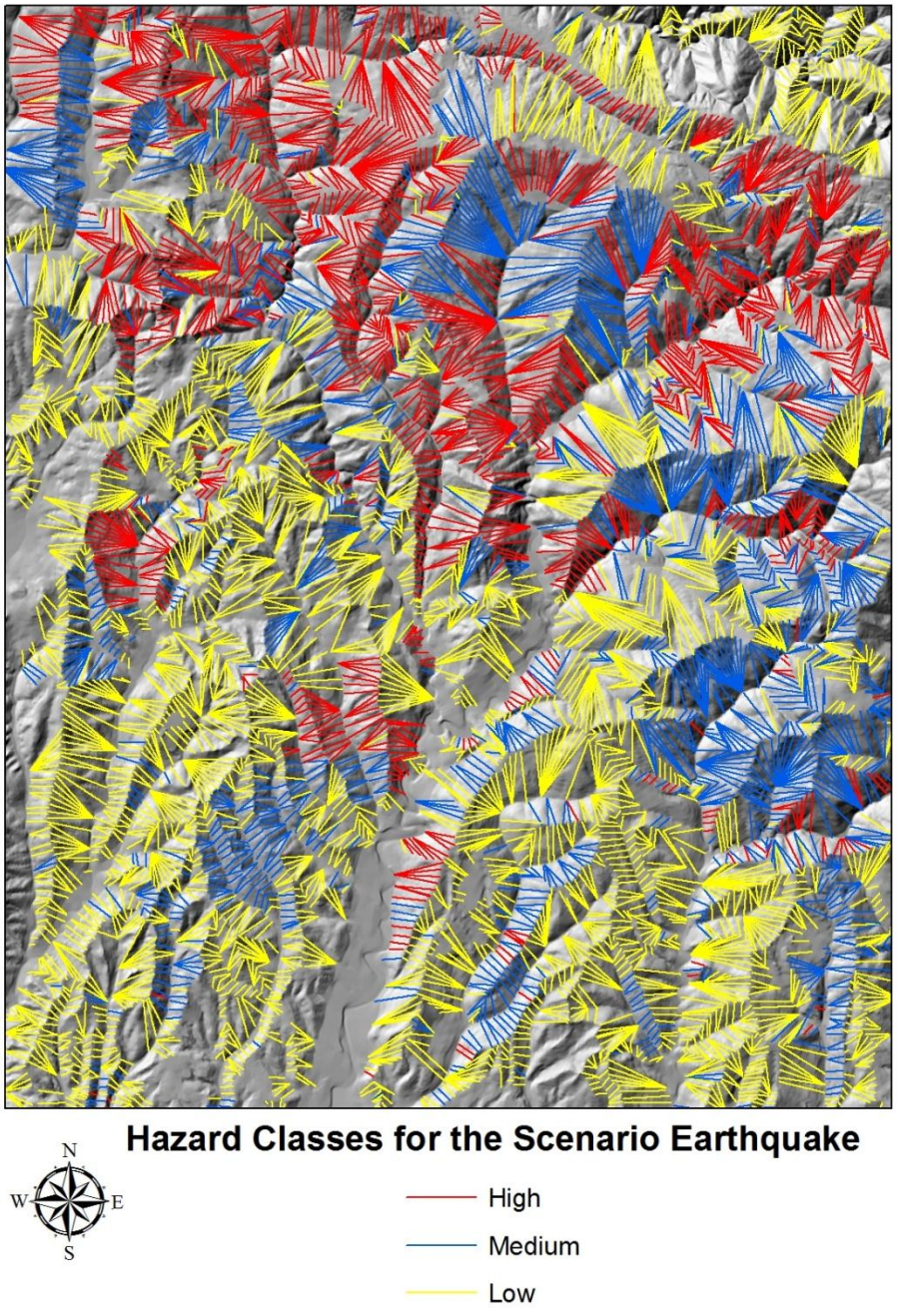


Figure 6.23 Predicted hazard classes for the scenario earthquake on SAS segment of San Andreas Fault (M_w=7.03)

CHAPTER 7

CONCLUSIONS AND RECOMMENDATIONS

7.1 Conclusions

An efficient and flexible framework is developed for the regional evaluation of earthquake-triggered coherent landslide hazards. This framework consists of reliable and efficient metamodels which replaces the computationally expensive static and dynamic slope stability analyses. The framework also includes methods for automatic delineation of slope profiles from digital elevation models. Based on the current study, the following conclusions were reached:

- Using the soft computing methodology developed in this study which uses a sequential metamodeling approach, reliable predictions of the performances of a large number of slopes with different geometric and material properties under different earthquake loads can be calculated in a matter of seconds. This time efficiency makes it possible to use this method in a regional seismic coherent landslide hazard analysis. This efficiency also makes it relatively easy to backcalculate the material shear strength parameters assigned to the slope profiles using their previous performances.
- Relatively simple but effective automated slope profile delineation methods are introduced. These methods utilize an extension to already available tools (Hydrology tools). An addition to these methods, a method to delineate slope profiles along highway sections is also developed. Using this extension, slope profile delineation method developed for natural slopes can be extended to manmade features.

- Using the Lower Schultheis Road East landslide case history, the performance of the SVM classifier was investigated by comparing the class calculated using FLAC analysis and the class predicted using SVM classifier. SVM classifier showed a very good performance predicting the hazard classes with given slope material and geometric properties, and intensity measures of the earthquake loading.
- The developed framework was evaluated using the landslide inventory of the 1989 Loma Prieta earthquake (Mw = 6.9) within the 7.5 min. Laurel, CA quadrangle. Even though the model predicted the locations of the landslides, it also over predicted some of the landslides. However, the shear strength values of these over predicted landslides can easily be calibrated using the metamodel. It has also been shown that the model can easily be used to predict the seismic performances during a future scenario earthquake.

7.2 Recommendations for Future Research

Even though the current study developed a reliable and efficient framework for evaluating the regional earthquake-triggered coherent landslide hazards, some improvement and further work are needed to enhance the current framework:

- Although a deterministic seismic landslide hazard analysis is performed for the pilot study, due to the flexibility and efficiency of predicting reliable slope performances, the soft computing based methods readily allow themselves to be included in a probabilistic landslide hazard analysis. The efficiency of the soft computing based methods permits the uncertainties in the prediction of earthquake ground shaking and material soil properties to be incorporated in a probabilistic framework. Moreover, because the soft computing based model is

also flexible in regards to the choice of input intensity measures (IM), it can easily be integrated into a performance-based design.

- There are a variety of geotechnical structures prone to seismically induced deep-seated coherent slope failures. These geotechnical structures can be natural slopes, or engineered slopes such as solid waste landfills, earth and rockfill dams, and embankments. Although the framework developed in this study is used for natural slopes, it can also easily be used for manmade slopes. Furthermore, one can argue that the homogenous simple slope model used for developing the metamodels in Chapter 3 and Chapter 4 would be a more suitable model for engineered slopes because of the more homogenous nature of these slopes.
- In the case of manmade slopes such as highway embankments, the methods developed for slope delineation in Chapter 5 can be used with higher resolution DEMs (LIDAR). Because of the relatively small size of these geostructures, 1/3 arc-sec (~10 m) DEM doesn't provide enough resolution to accurately extract slope profile information.
- Only dry slopes are analyzed because the study area was unusually dry at the time of the earthquake. However, if the ground water levels are high enough this may greatly affect the static and dynamic stability of slopes. An improved metamodel can be created using the ground water level as a design variable.
- Only homogenous slopes are modeled because it is common practice in regional landslide hazard analysis to assign average shear strength values to geological units due to lack of detailed material strength information. However, if detailed geological and geotechnical information is available, two or more layered profiles can be introduced into the metamodels by assigning the thickness and the shear strength parameters of layers as design variables.
- Only simple linear slope models are used in the process of creating metamodels. However, linear simple slope profile doesn't always satisfactorily model natural

slopes. Slopes which have irregular profiles may behave differently. In order to address this problem, different metamodels can be created by selecting a number of typical slope profiles considering plan and profile curvature.

- A constant depth to bedrock is assumed for all slopes, because there is not enough information regarding the depth to the bedrock. Parametric study in Chapter 6 showed that depth to bedrock doesn't have a significant impact on the crest displacement. But if the depth to bedrock is known, this parameter can easily be implemented as a design variable in the metamodeling process.
- Although a methodology is developed for delineating the slope profiles along highway sections in Chapter 5, the highway sections in Laurel quadrangle were not investigated. The resolution of the DEM for the study area was not high enough to properly delineate slope profiles along highway sections. However, the framework presented in this study is equally applicable to manmade slopes.
- Instead of using artificial earthquake records for creating metamodels, real earthquake records can be used by spectral matching.
- For the design of experiments, correlations between the design variables can be taken into consideration.
- Near fields earthquake records which have velocity pulses can be included in the metamodels building process.

APPENDIX A

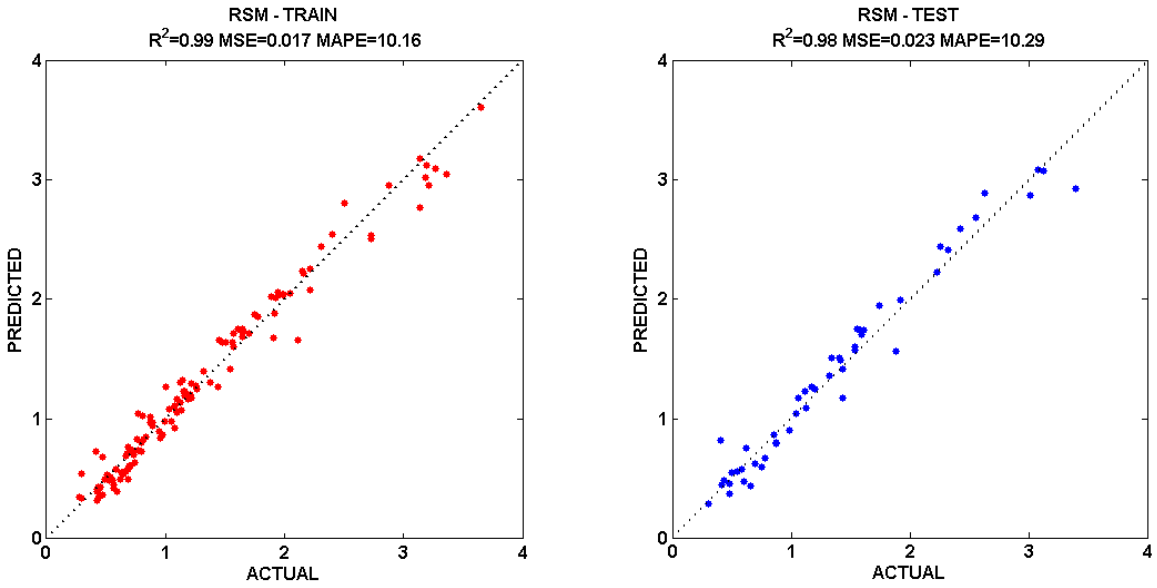


Figure A.1 Performance of RSM model for 120 learning points

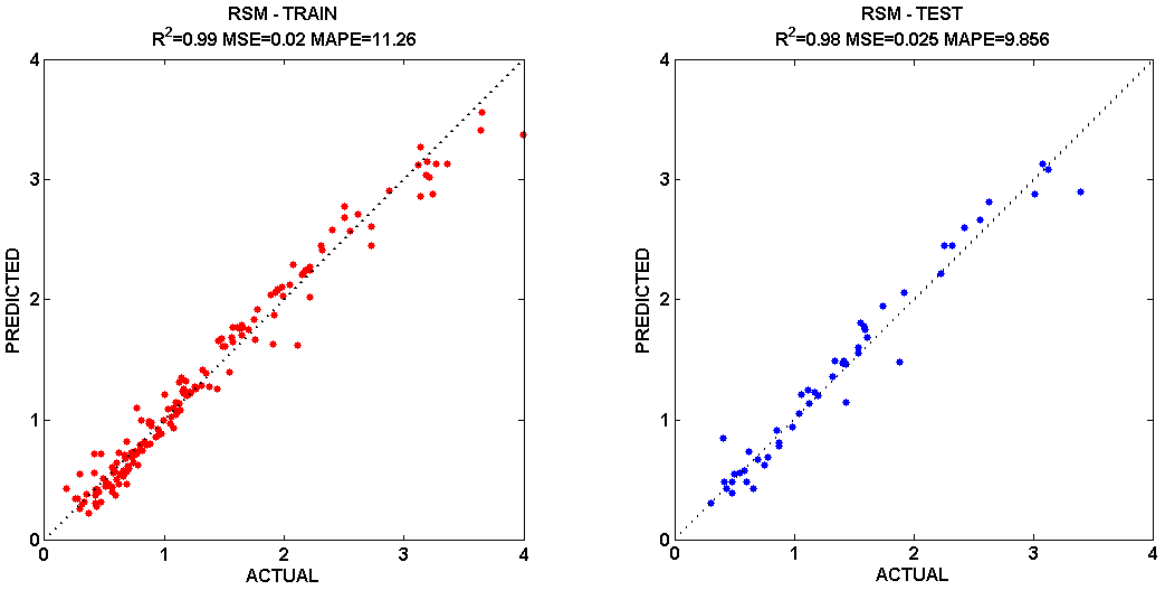


Figure A.2 Performance of RSM model for 160 learning points

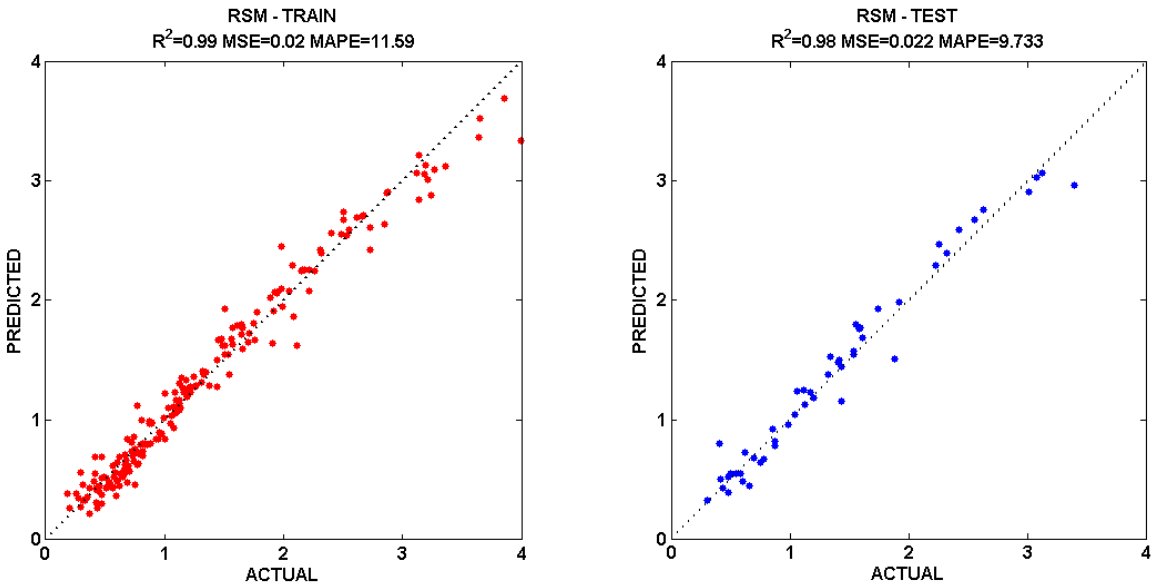


Figure A.3 Performance of RSM model for 200 learning points

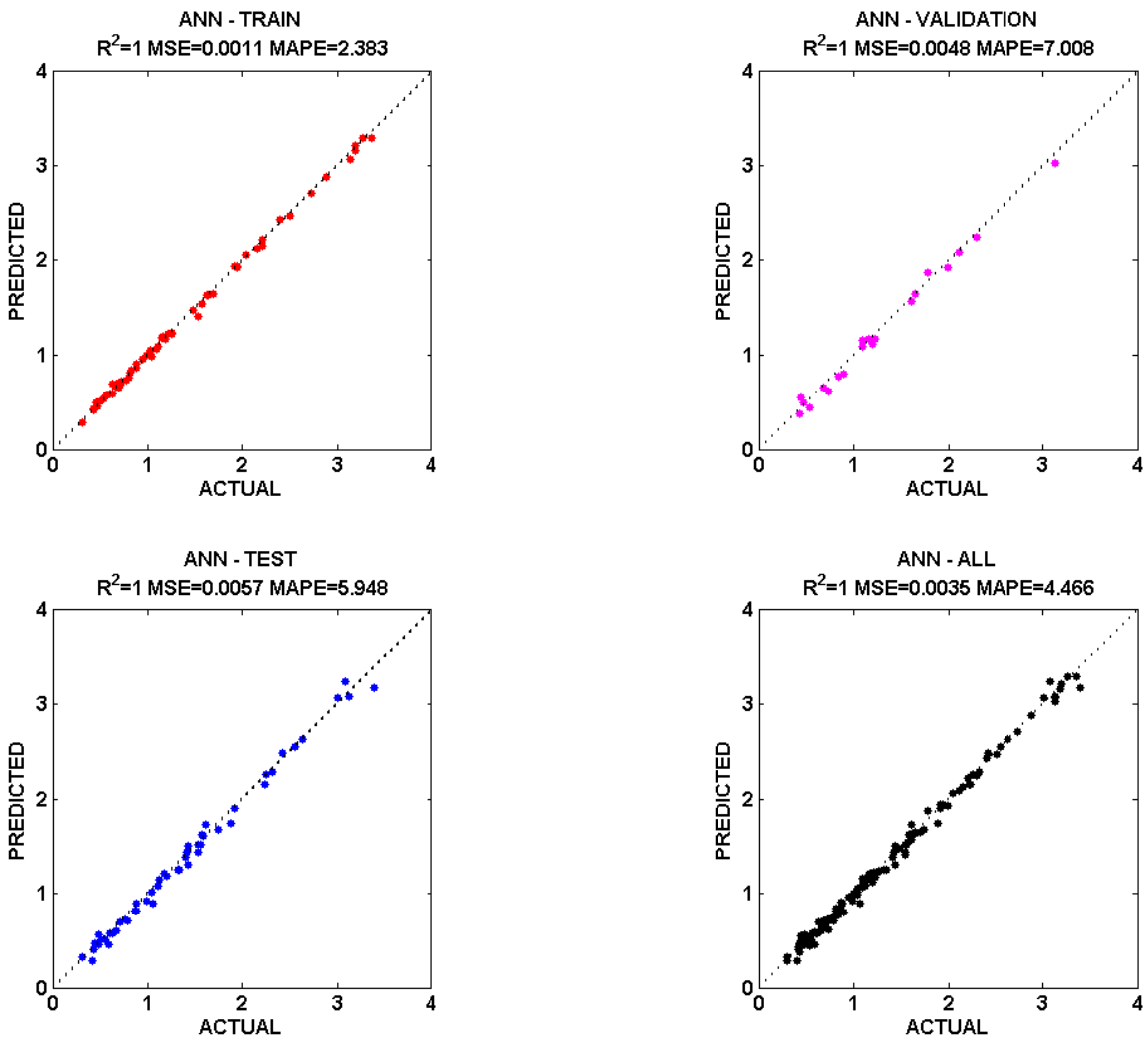


Figure A.4 Performance of ANN model for 80 learning points

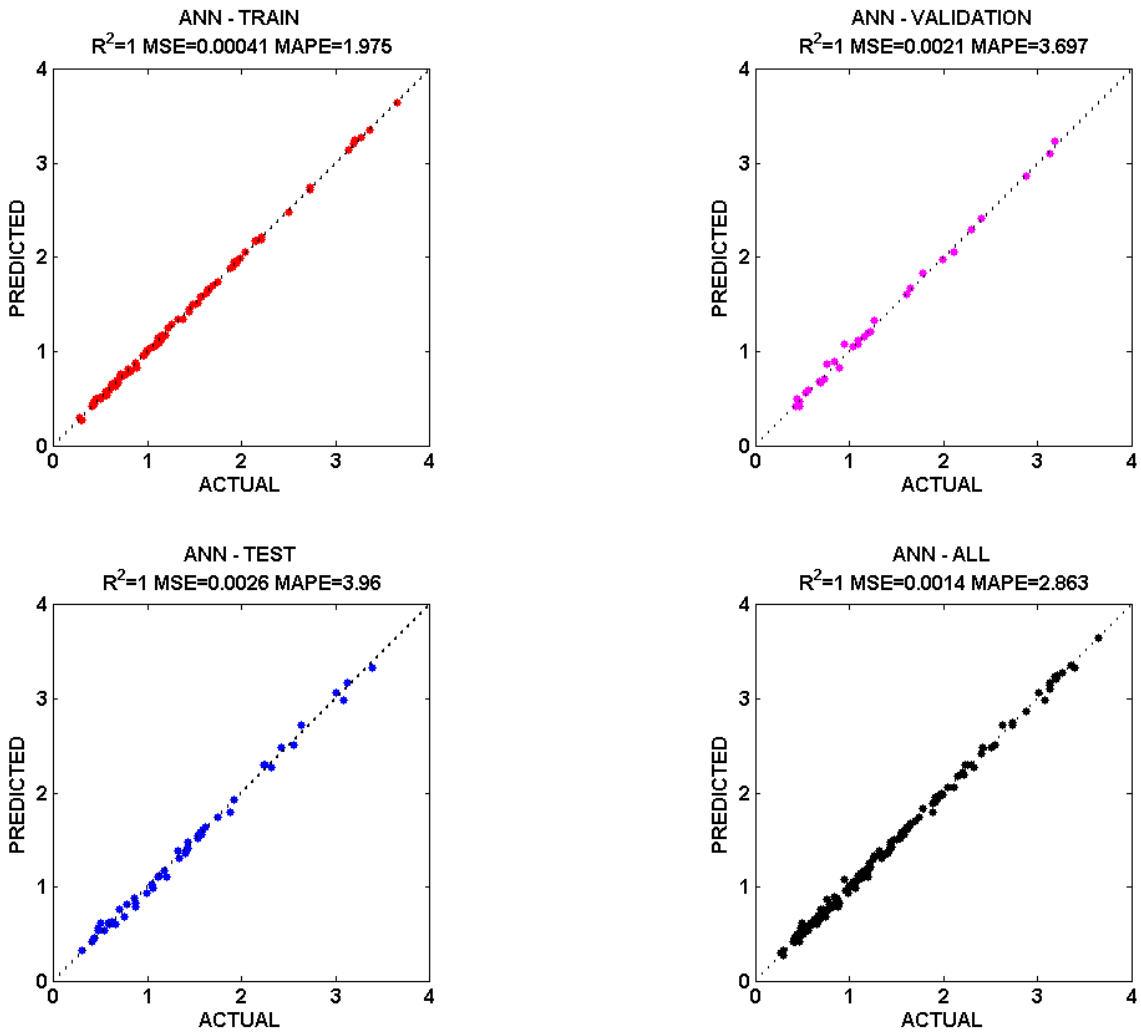


Figure A.5 Performance of ANN model for 120 learning points

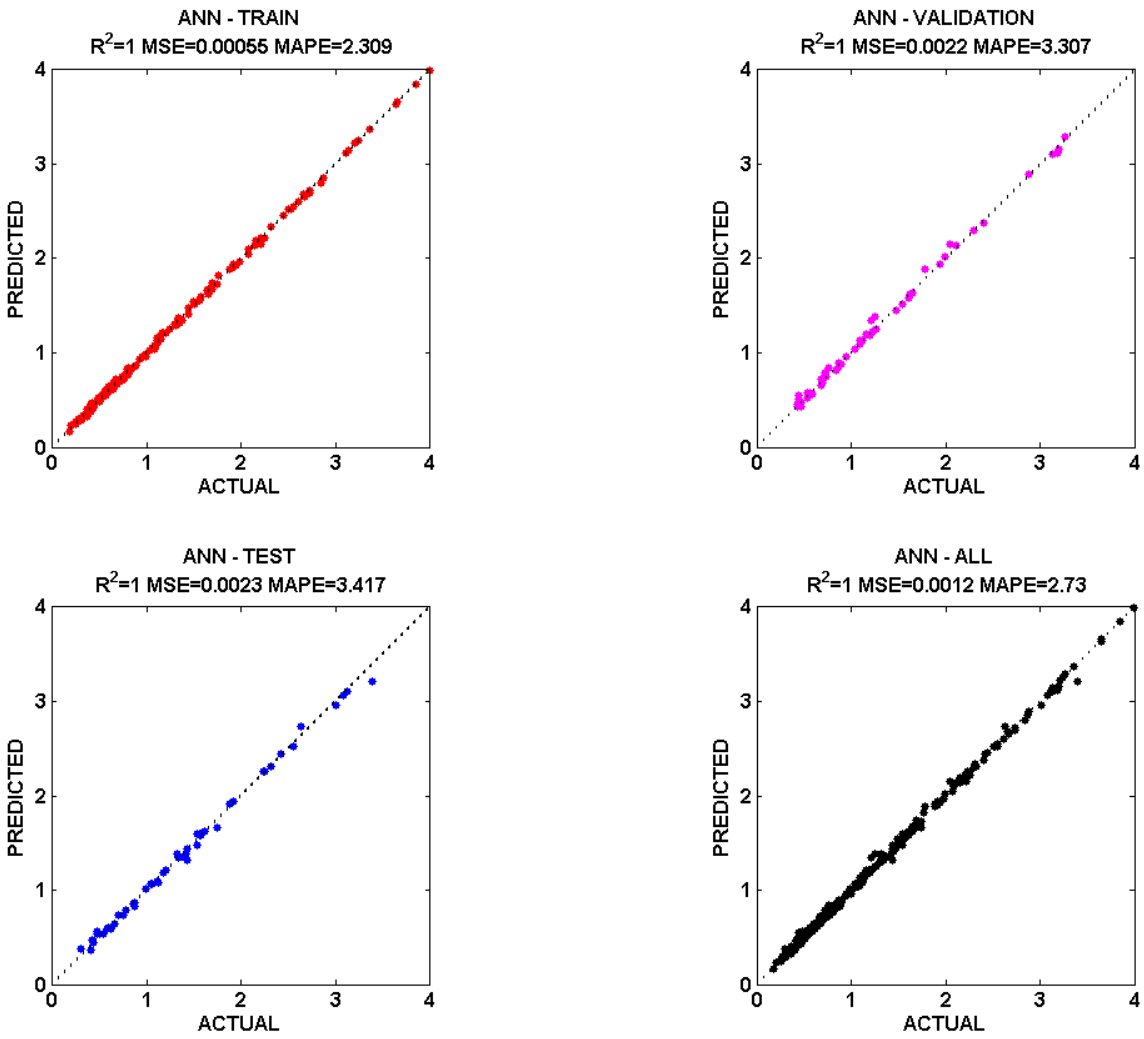


Figure A.6 Performance of ANN model for 200 learning points

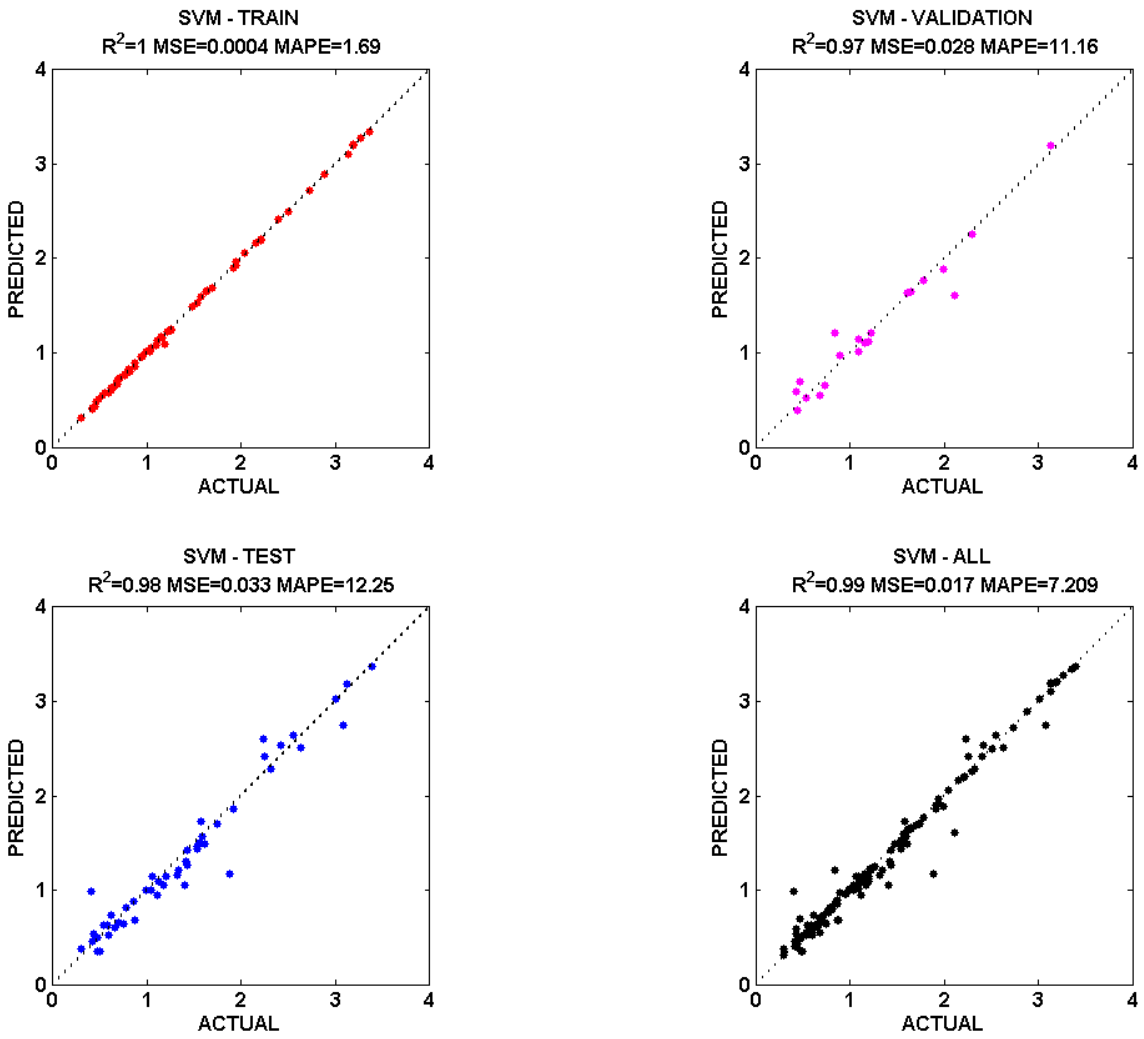


Figure A.7 Performance of SVM model for 80 learning points

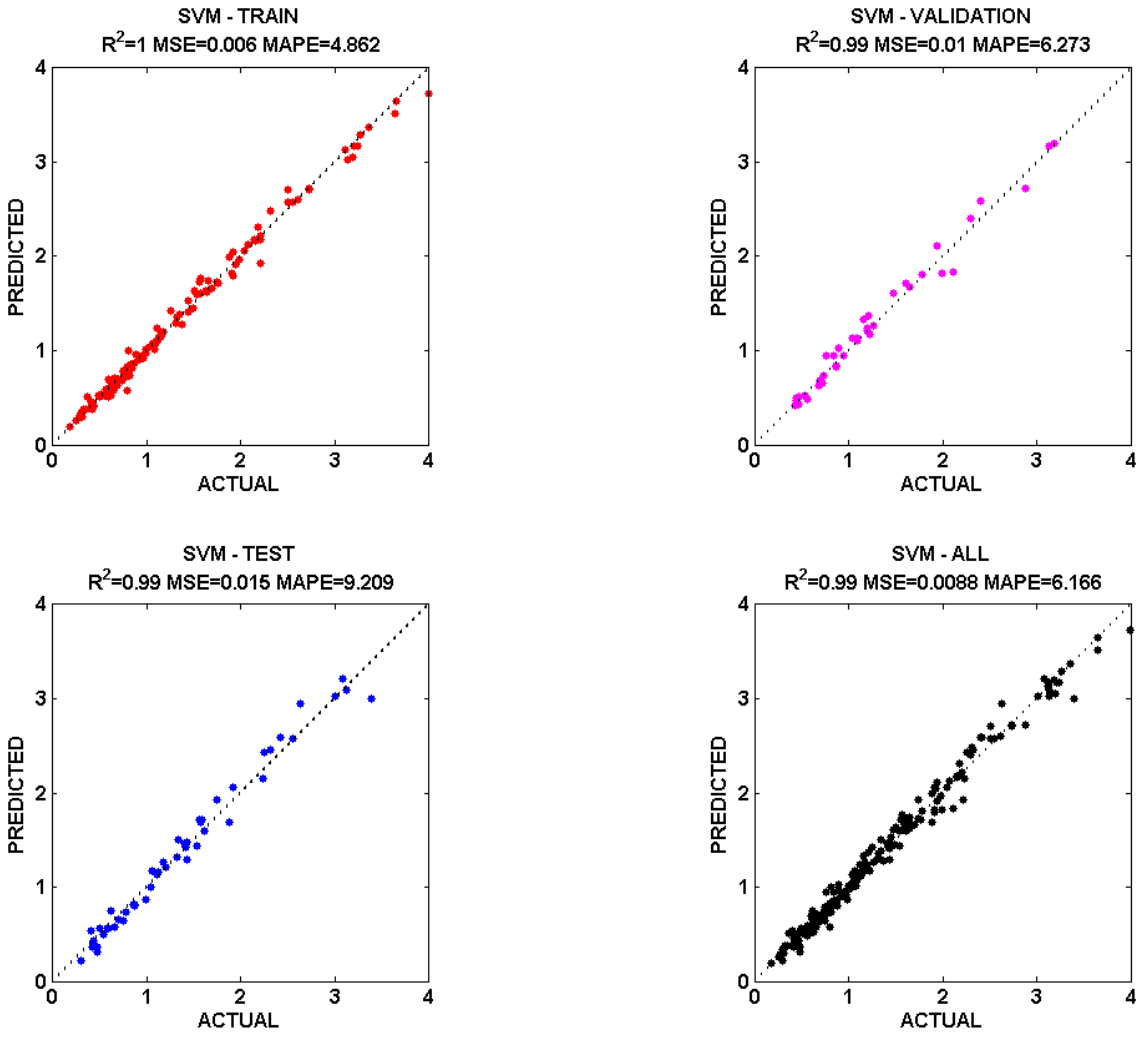


Figure A.8 Performance of SVM model for 160 learning points

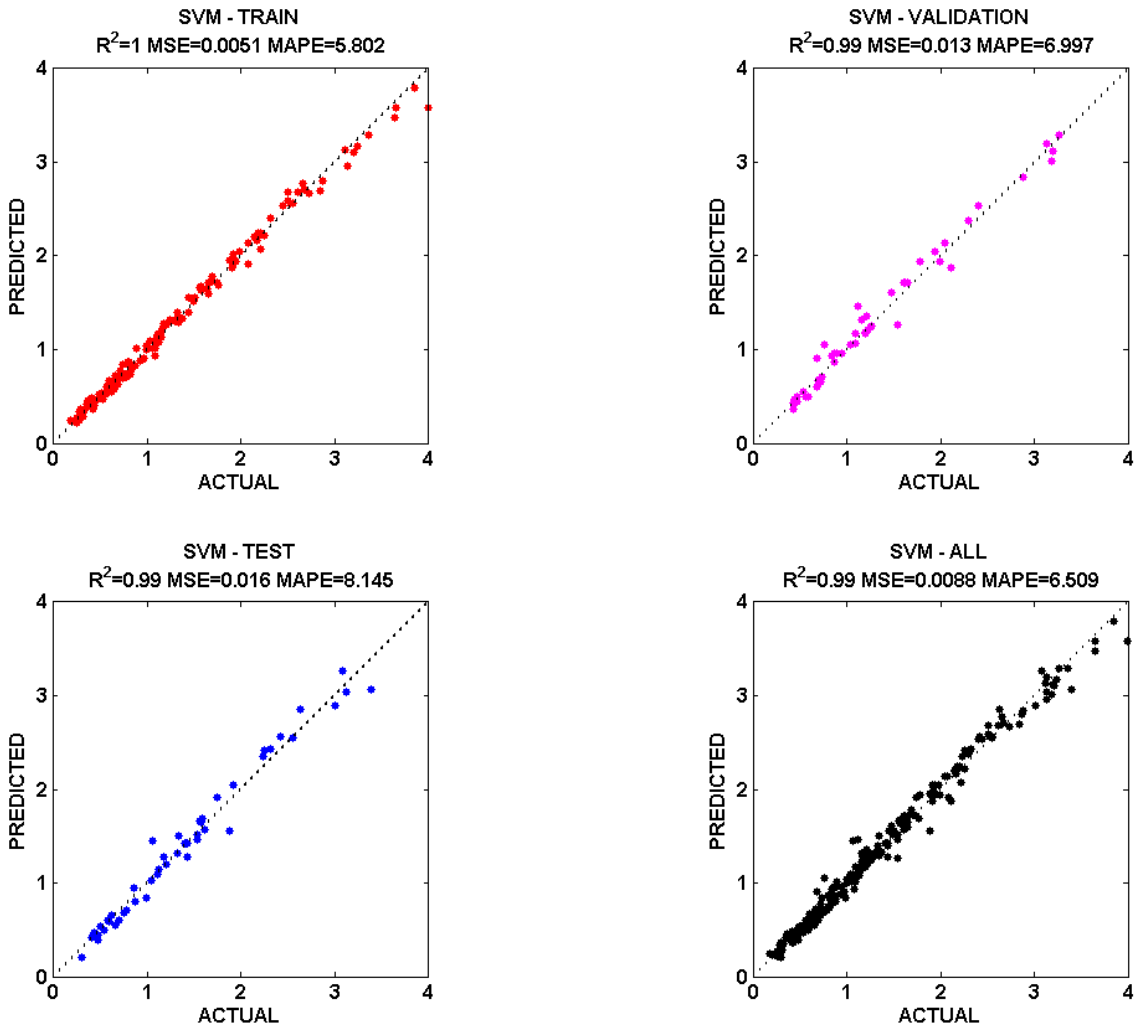


Figure A.9 Performance of SVM model for 200 learning points

Table A.1 Confusion matrices of Naïve Bayes classifier for 80 training and 50 test points

		PREDICTED U	PREDICTED S
ACTUAL	U	29	7
	S	7	37
		82.5%	

Training

		PREDICTED U	PREDICTED S
ACTUAL	U	16	3
	S	4	27
		86%	

Testing

Table A.2 Confusion matrices of Naïve Bayes classifier for 120 training and 50 test points

		PREDICTED U	PREDICTED S
ACTUAL	U	47	7
	S	10	56
		85.83%	

Training

		PREDICTED U	PREDICTED S
ACTUAL	U	17	2
	S	4	27
		88%	

Testing

Table A.3 Confusion matrices of Naïve Bayes classifier for 160 training and 50 test points

		PREDICTED U	PREDICTED S
ACTUAL	U	67	9
	S	10	74
		88.13%	

Training

		PREDICTED U	PREDICTED S
ACTUAL	U	17	2
	S	4	27
		88%	

Testing

Table A.4 Confusion matrices of ANN classifier for 80 learning and 50 test points

ACTUAL \ PREDICTED	U	S
U	28	0
S	2	30
	96.67%	

Training

ACTUAL \ PREDICTED	U	S
U	8	0
S	0	12
	100%	

Validation

ACTUAL \ PREDICTED	U	S
U	19	0
S	4	27
	92%	

Testing

Table A.5 Confusion matrices of ANN classifier for 160 learning and 50 test points

ACTUAL \ PREDICTED	U	S
U	56	1
S	3	60
	96.67%	

Training

ACTUAL \ PREDICTED	U	S
U	19	0
S	0	21
	100%	

Validation

ACTUAL \ PREDICTED	U	S
U	19	0
S	2	29
	96%	

Testing

Table A.6 Confusion matrices of ANN classifier for 200 learning and 50 test points

ACTUAL \ PREDICTED	U	S
U	70	0
S	1	79
	99.33%	

Training

ACTUAL \ PREDICTED	U	S
U	22	0
S	1	27
	98%	

Validation

ACTUAL \ PREDICTED	U	S
U	18	1
S	2	29
	94%	

Testing

Table A.7 Confusion matrices of SVM classifier for 120 training and 50 test points

ACTUAL \ PREDICTED	U	S
U	51	3
S	2	64
	95.83%	

Training

ACTUAL \ PREDICTED	U	S
U	19	0
S	0	31
	100%	

Testing

Table A.8 Confusion matrices of SVM classifier for 160 training and 50 test points

ACTUAL \ PREDICTED	U	S
U	76	0
S	0	84
	100%	

Training

ACTUAL \ PREDICTED	U	S
U	19	0
S	1	30
	98%	

Testing

Table A.9 Confusion matrices of SVM classifier for 200 training and 50 test points

ACTUAL \ PREDICTED	U	S
U	90	2
S	1	107
	98.5%	

Training

ACTUAL \ PREDICTED	U	S
U	19	0
S	0	31
	100%	

Testing

APPENDIX B

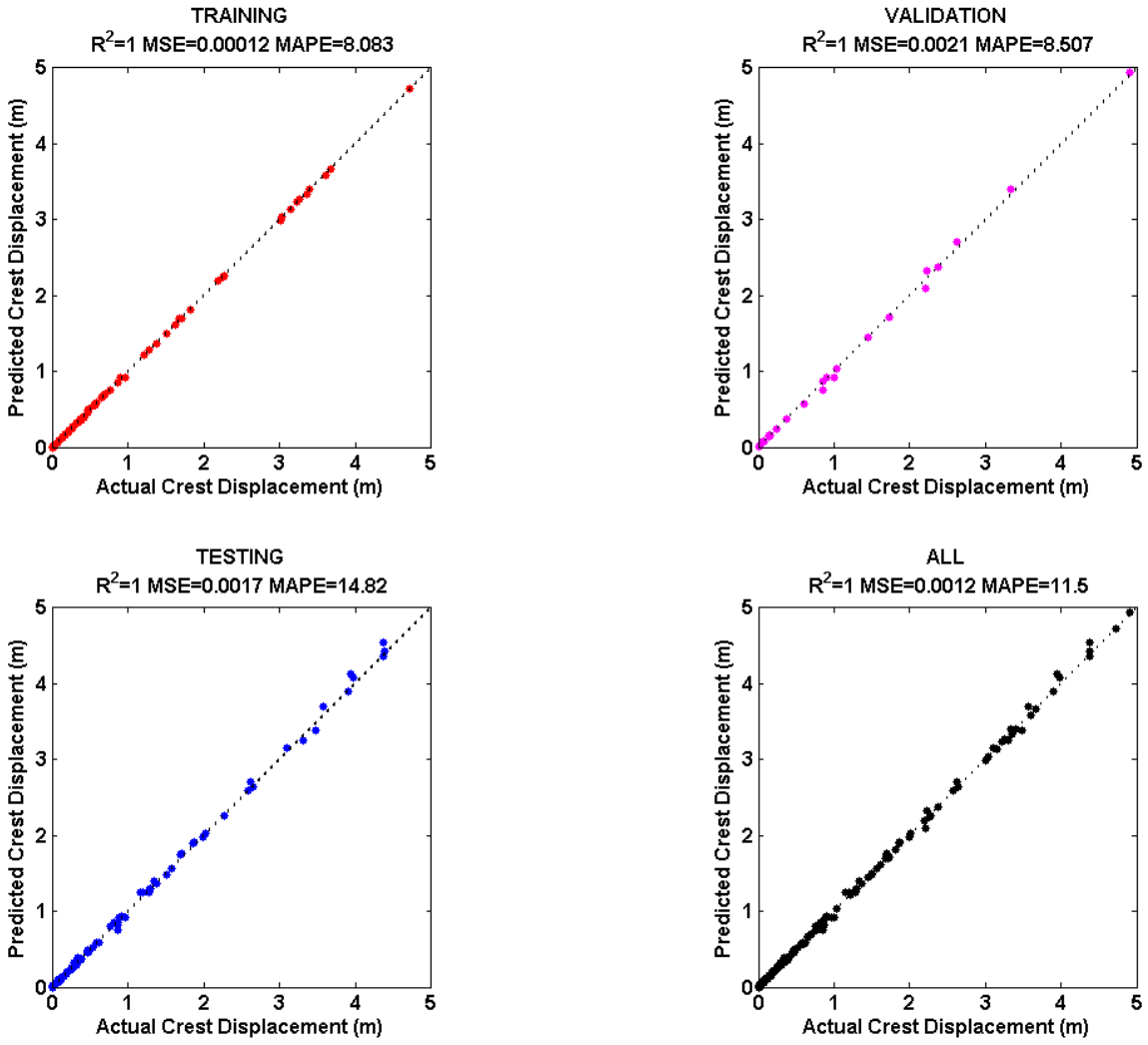


Figure B.1 Performance of ANN model for 100 learning points for single slope case

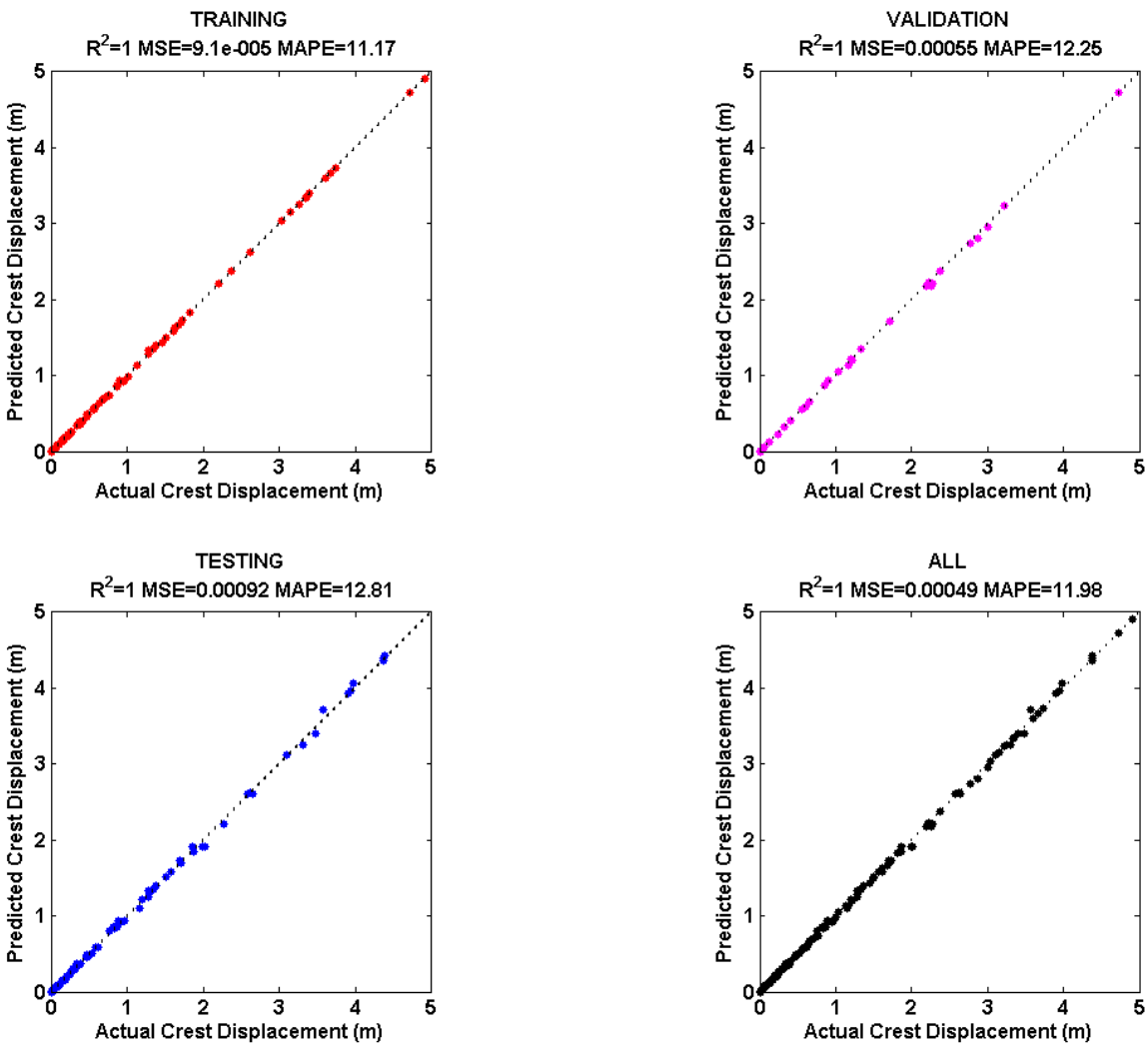


Figure B.2 Performance of ANN model for 150 learning points for single slope case

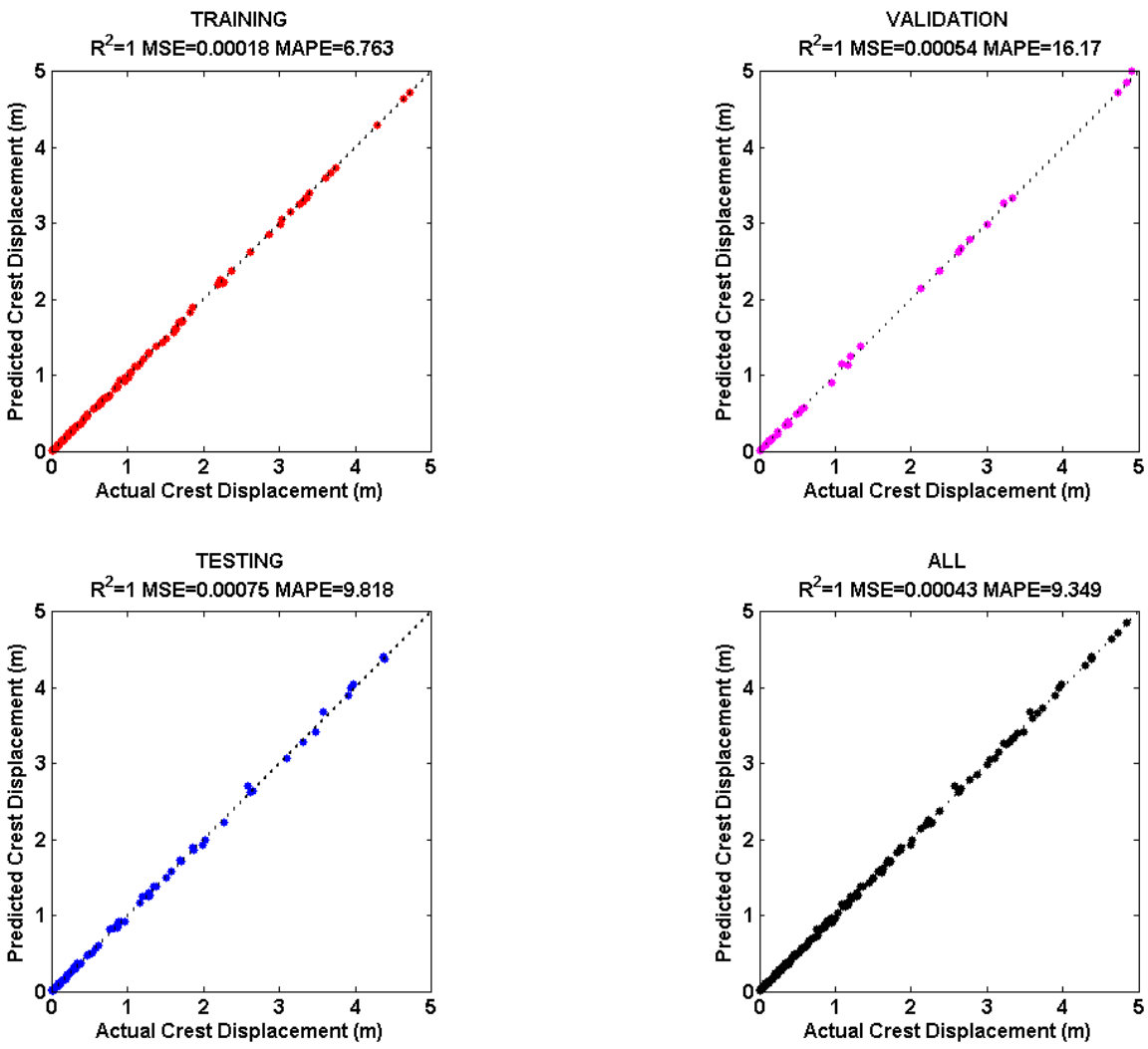


Figure B.3 Performance of ANN model for 200 learning points for single slope case

Table B.1 Confusion matrices of SVM model trained by 100 training points

ACTUAL \ PREDICTED	L	M	H
L	39	0	0
M	1	26	0
H	0	0	34
	99%		

Training

ACTUAL \ PREDICTED	L	M	H
L	31	6	1
M	4	21	6
H	0	7	24
	76%		

Testing

Table B.2 Confusion matrices of SVM model trained by 150 training points

ACTUAL \ PREDICTED	L	M	H
L	59	4	0
M	7	35	1
H	3	4	37
	87.3%		

Training

ACTUAL \ PREDICTED	L	M	H
L	35	3	0
M	5	20	6
H	0	7	24
	79%		

Testing

Table B.3 Confusion matrices of SVM model trained by 200 training points

ACTUAL \ PREDICTED	L	M	H
L	72	7	0
M	11	44	4
H	2	5	55
	85.5%		

Training

ACTUAL \ PREDICTED	L	M	H
L	35	3	0
M	7	19	5
H	0	4	27
	81%		

Testing

Table B.4 Confusion matrices of SVM model trained by 250 training points

ACTUAL \ PREDICTED	L	M	H
L	79	14	0
M	13	62	5
H	4	11	62
	81.2%		

Training

ACTUAL \ PREDICTED	L	M	H
L	34	4	0
M	5	21	5
H	0	4	27
	82%		

Testing

Table B.5 Confusion matrices of SVM model trained by 300 training points

ACTUAL \ PREDICTED	L	M	H
L	101	11	0
M	16	73	6
H	1	10	82
	85.3%		

Training

ACTUAL \ PREDICTED	L	M	H
L	35	3	0
M	5	20	6
H	0	3	28
	83%		

Testing

Table B.6 Confusion matrices of SVM model trained by 400 training points

ACTUAL \ PREDICTED	L	M	H
L	139	10	0
M	20	97	9
H	2	15	108
	86%		

Training

ACTUAL \ PREDICTED	L	M	H
L	35	3	0
M	5	20	6
H	0	2	29
	84%		

Testing

Table B.7 Confusion matrices of SVM model trained by 450 training points

PREDICTED

ACTUAL	L	M	H
L	148	15	0
M	24	113	7
H	3	16	124
	85.6%		

Training

PREDICTED

ACTUAL	L	M	H
L	35	3	0
M	5	21	5
H	0	3	28
	84%		

Testing

APPENDIX C

Table C.1 Visual Basic Code for Creating Points along Ridge Lines

```
Public Sub CreatePointsAlongRidgeLines ()
  'Creates points along ridge lines at a user defined interval
  'Select Layer with the watershed polygons in the Table of Contents window
  'This routine store created points in the empty point feature class at top of TOC

  Dim pMxDoc As IMxDocument
  Dim pMap As IMap
  Dim pInGeometry As IGeometry
  Dim pInLayer As ILayer
  Dim pInFLayer As IFeatureLayer
  Dim pOutFLayer As IFeatureLayer
  Dim pInFCursor As IFeatureCursor
  Dim pOutFCursor As IFeatureCursor
  Dim pOutFBuffer As IFeatureBuffer
  Dim pInFClass As IFeatureClass
  Dim pOutFClass As IFeatureClass
  Dim pSelSet As ISelectionSet
  Dim pFSelection As IFeatureSelection
  Dim pInFeature As IFeature
  Dim pCurve As ICurve
  Dim pPointCollection As IPointCollection
  Dim pConstructMultipoint As IConstructMultipoint

  Set pMxDoc = ThisDocument
  Set pMap = pMxDoc.FocusMap
  Set pInLayer = pMxDoc.SelectedLayer
  Set pOutFLayer = pMap.Layer(0)
  Set pInFClass = pInFLayer.FeatureClass
  Set pOutFClass = pOutFLayer.FeatureClass
  Set pFSelection = pInFLayer
  Set pSelSet = pFSelection.SelectionSet

  'Ask user for distance between points
  Dim pPointDist As Double
  pPointDist = InputBox("Enter the Distance between points: ")

  Set pOutFBuffer = pOutFClass.CreateFeatureBuffer
  Set pOutFCursor = pOutFClass.Insert(True)

  If pSelSet.Count <> 0 Then
    pFSelection.SelectionSet.Search Nothing, True, pInFCursor
  Else
    Set pInFCursor = pInFClass.Search(Nothing, True)
  End If
  Set pInFeature = pInFCursor.NextFeature
    Do While Not pInFeature Is Nothing
    Set pInGeometry = pInFeature.Shape
    Set pCurve = pInGeometry
    Set pConstructMultipoint = New Multipoint
```

```
    pConstructMultipoint.ConstructDivideLength pCurve, pPointDist
    Set pPointCollection = pConstructMultipoint
    Dim i As Long
    For i = 0 To pPointCollection.PointCount - 1
      Set pOutFBuffer.Shape = pPointCollection.Point(i)
      pOutFCursor.InsertFeature pOutFBuffer
      k = k + 1
    Next i
  Set pInFeature = pInFCursor.NextFeature
  Loop
End Sub
```

Table C.2 Visual Basic Code for Creating Slope Profile Polylines (Slope Unit Approach)

```
Const pi As Double = 3.14159265
Public Sub CreateSlopeProfiles()
Dim pMxDoc As IMxDocument
Dim pFtrLyr As IFeatureLayer
Dim pPolyFtrLyr As IFeatureLayer
Dim pPolyFtrCls As IFeatureClass
Dim pPtFtrCls As IFeatureClass
Dim pPolyFtrCsr As IFeatureCursor
Dim pPtFtrCsr As IFeatureCursor
Dim pPolyFtr As IFeature
Dim pPtFtr As IFeature
Dim pSpFltr As ISpatialFilter

   ' Setup the spatial filter for intersections
   Set pSpFltr = New SpatialFilter
   pSpFltr.SpatialRel = esriSpatialRelIntersects

   ' Get a ref to the point layer
   Set pMxDoc = ThisDocument
   Set pFtrLyr = pMxDoc.FocusMap.Layer(0)
   Set pPtFtrCls = pFtrLyr.FeatureClass

   ' Get a ref to the polygon layer
   Set pPolyFtrLyr = pMxDoc.FocusMap.Layer(1)
   Set pPolyFtrCls = pPolyFtrLyr.FeatureClass

   ' Get a cursor on all the polygon features
   ' and loop thru them
   Set pPolyFtrCsr = pPolyFtrCls.Search(Nothing, False)
   Set pPolyFtr = pPolyFtrCsr.NextFeature
   While Not pPolyFtr Is Nothing

      ' Get a cursor on all the points that fall within this polygon
      ' and loop thru them
      Set pSpFltr.Geometry = pPolyFtr.ShapeCopy
      Set pPtFtrCsr = pPtFtrCls.Search(pSpFltr, False)
      Set pPtFtr = pPtFtrCsr.NextFeature
      While Not pPtFtr Is Nothing

      ' Call the routine to extendlines
      Extendlines pPolyFtr.ShapeCopy, pPtFtr.ShapeCopy, pPtFtr.Value(2)

      Set pPtFtr = pPtFtrCsr.NextFeature

      Wend

      Set pPolyFtr = pPolyFtrCsr.NextFeature

   Wend

   ' Get a ref to the polyline layer
   Set pFtrLyr = pMxDoc.FocusMap.Layer(2)
   Dim pPlineFtrCls As IFeatureClass
```

```
    Set pPlineFtrCls = pFtrLyr.FeatureClass

    Dim pFeatureClassManage As IFeatureClassManage
    Set pFeatureClassManage = pPlineFtrCls
    pFeatureClassManage.UpdateExtent

End Sub

Private Sub Extendlines(pPolygon As IPolygon, pPt As IPoint, Angle As Double)

Dim pTopOp As ITopologicalOperator
Dim pBndry As IPolyline
Dim pToPt As IPoint
Dim pPline As IPolyline
Dim pConsCurve As IConstructCurve
Dim dAng As Double
Dim pTrans2D As ITransform2D
Dim bOK As Boolean

    ' Get the boundary of the polygon
    Set pTopOp = pPolygon
    Set pBndry = pTopOp.Boundary

    Dim pMxDoc As IMxDocument
    Set pMxDoc = ThisDocument

    'Dim pPageLayout As IPageLayout
    'Set pPageLayout = pMxDoc.PageLayout

    'Dim pPage As IPage
    'Set pPage = pPageLayout.Page

    ' Get a ref to the polyline layer
    Dim pFtrLyr As IFeatureLayer
    Set pFtrLyr = pMxDoc.FocusMap.Layer(2)
    Dim pPlineFtrCls As IFeatureClass
    Set pPlineFtrCls = pFtrLyr.FeatureClass

Dim pi
pi = 3.14159265
If Angle = 1 Then
dAng = 0
End If
If Angle = 2 Then
dAng = 2 * pi * 7 / 8
End If
If Angle = 4 Then
dAng = 2 * pi * 6 / 8
End If
If Angle = 8 Then
dAng = 2 * pi * 5 / 8
End If
If Angle = 16 Then
dAng = 2 * pi * 4 / 8
End If
If Angle = 32 Then
```

```
dAng = 2 * pi * 3 / 8
End If
If Angle = 64 Then
dAng = 2 * pi * 2 / 8
End If
If Angle = 128 Then
dAng = 2 * pi * 1 / 8
End If
        ' create a small line from the point to 0 degrees (due east)
        'dAng = Angle
        Set pPline = New Polyline
        pPline.FromPoint = pPt
        Set pToPt = New Point
        pToPt.PutCoords pPt.X + 0.0001, pPt.Y
        'pToPt.PutCoords pPt.X + 10, pPt.Y
        pPline.ToPoint = pToPt
        ' rotate the line
        Set pTrans2D = pPline
        'changed the line below so that 0 is north, 90 is east, etc.
        pTrans2D.Rotate pPt, dAng
        ' extend the line to the boundary
        Set pConsCurve = New Polyline
        pConsCurve.ConstructExtended pPline, pBndry, 8, bOK
        Set pPline = pConsCurve

If pPline.Length > 20 Then
Set pConsCurve = New Polyline
pConsCurve.ConstructExtended pPline, pBndry, 0, bOK
Set pPline = pConsCurve

    'create a new feature for the polyline layer and add the new line to this layer
    Dim pPlineFtr As IFeature
    Set pPlineFtr = pPlineFtrCls.CreateFeature
    Set pPlineFtr.Shape = pPline
    pPlineFtr.Store

End If
End Sub
```

Table C.3 Visual Basic Code for Creating Slope Profile Polylines (Steepest Descent Approach)

```
Public Sub CreateSlopeProfiles()

Dim pMxDoc As IMxDocument
Dim pPointFtrLyr As IFeatureLayer
Dim pPolygonFtrLyr As IFeatureLayer
Dim pPointFtrCls As IFeatureClass
Dim pPolygonFtrCls As IFeatureClass
Dim pPointFtrCsr As IFeatureCursor
Dim pPolygonFtrCsr As IFeatureCursor
Dim pPointFtr As IFeature
Dim pPolygonFtr As IFeature
Dim pSpFltr As ISpatialFilter
Dim pLayer As ILayer
Set pMxDoc = ThisDocument

   ' Setup the spatial filter for intersections
   Set pSpFltr = New SpatialFilter
   pSpFltr.SpatialRel = esriSpatialRelIntersects

   ' Get a ref to the point layer
   Set pPointFtrLyr = pMxDoc.FocusMap.Layer(0)
   Set pPointFtrCls = pPointFtrLyr.FeatureClass

   ' Get a ref to the polygon layer
   Set pPolygonFtrLyr = pMxDoc.FocusMap.Layer(1)
   Set pPolygonFtrCls = pPolygonFtrLyr.FeatureClass

   ' Get a ref to distance and direction raster (Use flowdirection raster for both!)
   Dim pDistanceRaster As IRaster
   Dim pDirectionRaster As IRaster
   Dim pRasLayer As IRasterLayer
   Set pLayer = pMxDoc.FocusMap.Layer(2)
   Set pRasLayer = pLayer
   Set pDistanceRaster = pRasLayer.Raster
   Set pDirectionRaster = pRasLayer.Raster

   ' Get a cursor on all the polygon features and loop thru them
   Set pPolygonFtrCsr = pPolygonFtrCls.Search(Nothing, False)
   Set pPolygonFtr = pPolygonFtrCsr.NextFeature
   'MsgBox pPolygonFtr.Value(0)
   While Not pPolygonFtr Is Nothing

      ' Get a cursor on all the points that fall within this polygon and loop thru them
      Set pSpFltr.Geometry = pPolygonFtr.ShapeCopy
      Set pPointFtrCsr = pPointFtrCls.Search(pSpFltr, False)
      Set pPointFtr = pPointFtrCsr.NextFeature
      'MsgBox pPointFtr.Value(0)

      While Not pPointFtr Is Nothing

      ' Create Flow Path Trace polylines
```

```
        Dim pPoly As IPolyline
        Set pPoly = New Polyline
        Dim pPolys As IPolyline
        Set pPolys = New Polyline
        Dim pGeometry As IGeometry
        Dim pPt As IPoint
        Set pPt = pPointFtr.Shape
        Set pGeometry = pPointFtr.Shape
        Set pPoint = pGeometry
        Dim pPointC As IPointCollection
        Set pPointC = New Multipoint
        pPointC.AddPoint pPoint
        'MsgBox pPointC.PointCount

         'Create a RasterDistanceOp operator
        Dim pGeomCol As IGeometryCollection
        Dim pDistanceOp As IDistanceOp
        Set pDistanceOp = New RasterDistanceOp
        Set pGeomCol = pDistanceOp.CostPathAsPolyline(pPointC, pDistanceRaster,
pDirectionRaster)
        'MsgBox pGeomCol.GeometryCount
        Set pPoly = pGeomCol.Geometry(0)
        Set pPolys = pGeomCol.Geometry(0)

' Intersection
Dim pTopoOp As esriGeometry.ITopologicalOperator
Set pTopoOp = pPoly
Dim pTopoOpI As esriGeometry.ITopologicalOperator
Set pTopoOpI = pPolygonFtr.ShapeCopy
Dim pBndry As IPolyline
Set pBndry = pTopoOpI.Boundary
Dim pGeomResult As IGeometry
Set pGeomResult = pTopoOp.Intersect(pBndry, esriGeometry0Dimension)

'MsgBox pGeomResult.IsEmpty
'MsgBox pGeomResult.GeometryType
If pGeomResult.IsEmpty = False Then

If TypeOf pGeomResult Is IPointCollection Then
    Dim pPointColl As esriGeometry.IPointCollection
    Set pPointColl = pGeomResult
    'MsgBox pPointColl.PointCount
    If pPointColl.PointCount >= 1 Then
    Set pGeomResult = pPointColl.Point(0)
    End If
  End If

  'MsgBox pGeomResult.IsEmpty
  'MsgBox pGeomResult.Dimension
  'MsgBox pGeomResult.GeometryType

Dim pPline As IPolyline
Dim pPointI As IPoint
Set pPointI = pGeomResult

'MsgBox pPointI.IsEmpty
```

```
'MsgBox pPointI.X
'MsgBox pPointI.Y

Dim pToPt As IPoint
Dim pPlineII As IPolyline

Dim pConsCurve As IConstructCurve
Dim bOK As Boolean

 ' create line from the start point to the end point
       Set pPlineII = New Polyline
       pPlineII.FromPoint = pPt
       pPlineII.ToPoint = pPointI
       'Set pPline = pPlineII

  ' extend the line to the boundary
       Set pConsCurve = New Polyline
       pConsCurve.ConstructExtended pPlineII, pBndry, 8, bOK
       Set pPline = pConsCurve

End If

If pPline.Length > 30 Then
       'extend the line to the boundary
       Set pConsCurve = New Polyline
       pConsCurve.ConstructExtended pPlineII, pBndry, 0, bOK
       Set pPline = pConsCurve

' Get a ref to the polyline layer
   Dim pNFtrLyr As IFeatureLayer
   Set pNFtrLyr = pMxDoc.FocusMap.Layer(3)
   Dim pNLineFtrCls As IFeatureClass
   Set pNLineFtrCls = pNFtrLyr.FeatureClass
'create a new feature for the polyline layer and add the new line to this layer
   Dim pNLineFtr As IFeature
   Set pNLineFtr = pNLineFtrCls.CreateFeature
   Set pNLineFtr.Shape = pPline
   pNLineFtr.Store

End If
Set pPointFtr = pPointFtrCsr.NextFeature
Wend
Set pPolygonFtr = pPolygonFtrCsr.NextFeature
Wend
End Sub
```

Table C.4 Visual Basic Code for Creating Attribute Tables for Slope Profile Polylines

```
Public Sub AddAttributes()

Dim pMxDoc As IMxDocument
Dim pPolylineFtrLyr As IFeatureLayer
Dim pPolylineFtrCls As IFeatureClass
Dim pPolylineFtrCsr As IFeatureCursor
Dim pPolylineFtr As IFeature
Set pMxDoc = ThisDocument

    ' Get a ref to the polyline layer
    Set pPolylineFtrLyr = pMxDoc.FocusMap.Layer(0)
    Set pPolylineFtrCls = pPolylineFtrLyr.FeatureClass

' Define and Add the FIELDS
Dim F_X As IFieldEdit
Dim F_Y As IFieldEdit
Dim F_Z As IFieldEdit
Dim T_X As IFieldEdit
Dim T_Y As IFieldEdit
Dim T_Z As IFieldEdit
Dim M_X As IFieldEdit
Dim M_Y As IFieldEdit
Dim M_Z As IFieldEdit
Dim Height As IFieldEdit
Dim Base As IFieldEdit
Dim Slope As IFieldEdit

Set F_X = New Field
Set F_Y = New Field
Set F_Z = New Field
Set T_X = New Field
Set T_Y = New Field
Set T_Z = New Field
Set M_X = New Field
Set M_Y = New Field
Set M_Z = New Field
Set Height = New Field
Set Base = New Field
Set Slope = New Field

With F_X
.Type = 3
.Name = "F_X"
End With
With F_Y
.Type = 3
.Name = "F_Y"
End With
With F_Z
.Type = 3
.Name = "F_Z"
End With
With T_X
```

```
.Type = 3
.Name = "T_X"
End With
With T_Y
.Type = 3
.Name = "T_Y"
End With
With T_Z
.Type = 3
.Name = "T_Z"
End With
With M_X
.Type = 3
.Name = "M_X"
End With
With M_Y
.Type = 3
.Name = "M_Y"
End With
With M_Z
.Type = 3
.Name = "M_Z"
End With
With Height
.Type = 3
.Name = "Height"
End With
With Base
.Type = 3
.Name = "Base"
End With
With Slope
.Type = 3
.Name = "Slope"
End With

pPolylineFtrCls.AddField F_X
pPolylineFtrCls.AddField F_Y
pPolylineFtrCls.AddField F_Z
pPolylineFtrCls.AddField T_X
pPolylineFtrCls.AddField T_Y
pPolylineFtrCls.AddField T_Z
pPolylineFtrCls.AddField M_X
pPolylineFtrCls.AddField M_Y
pPolylineFtrCls.AddField M_Z
pPolylineFtrCls.AddField Height
pPolylineFtrCls.AddField Base
pPolylineFtrCls.AddField Slope

' Find the index of the Fields
Dim indexFX As Long
indexFX = pPolylineFtrCls.FindField("F_X")
Dim indexFY As Long
indexFY = pPolylineFtrCls.FindField("F_Y")
Dim indexFZ As Long
indexFZ = pPolylineFtrCls.FindField("F_Z")
```

```
Dim indexTX As Long
indexTX = pPolylineFtrCls.FindField("T_X")
Dim indexTY As Long
indexTY = pPolylineFtrCls.FindField("T_Y")
Dim indexTZ As Long
indexTZ = pPolylineFtrCls.FindField("T_Z")
Dim indexMX As Long
indexMX = pPolylineFtrCls.FindField("M_X")
Dim indexMY As Long
indexMY = pPolylineFtrCls.FindField("M_Y")
Dim indexMZ As Long
indexMZ = pPolylineFtrCls.FindField("M_Z")
Dim indexHeight As Long
indexHeight = pPolylineFtrCls.FindField("Height")
Dim indexBase As Long
indexBase = pPolylineFtrCls.FindField("Base")
Dim indexSlope As Long
indexSlope = pPolylineFtrCls.FindField("Slope")

' Get a cursor on all the polyline features and loop thru them
Set pPolylineFtrCsr = pPolylineFtrCls.Search(Nothing, False)
Set pPolylineFtr = pPolylineFtrCsr.NextFeature
'MsgBox pPolylineFtr.Value(0)
While Not pPolylineFtr Is Nothing

Dim pFromPoint As IPoint
Dim pToPoint As IPoint
Dim pCurve As ICurve
Set pCurve = pPolylineFtr.Shape
Set pFromPoint = pCurve.FromPoint
Set pToPoint = pCurve.ToPoint

'Calculate From Point Attributes
Dim dXFrom As Double
Dim dYFrom As Double
Dim dZFrom As Double
dXFrom = pFromPoint.X
dYFrom = pFromPoint.Y
dZFrom = pFromPoint.Z
pPolylineFtr.Value(indexFX) = dXFrom
pPolylineFtr.Value(indexFY) = dYFrom
pPolylineFtr.Value(indexFZ) = dZFrom

'Calculate To Point Attributes
Dim dXTo As Double
Dim dYTo As Double
Dim dZTo As Double
dXTo = pToPoint.X
dYTo = pToPoint.Y
dZTo = pToPoint.Z
pPolylineFtr.Value(indexTX) = dXTo
pPolylineFtr.Value(indexTY) = dYTo
pPolylineFtr.Value(indexTZ) = dZTo

'Calculate Middle Point Attributes
Dim pMiddlePoint As IPoint
```

```
Dim dXMiddle As Double
Dim dYMiddle As Double
Dim dZMiddle As Double
Dim dDistance As Double
Dim bAsRatio As Boolean

Set pMiddlePoint = New Point
   pCurve.QueryPoint 0, dDistance, bAsRatio, pMiddlePoint
   dXMiddle = pMiddlePoint.X
   dYMiddle = pMiddlePoint.Y
   dZMiddle = pMiddlePoint.Z
pPolylineFtr.Value(indexMX) = dXMiddle
pPolylineFtr.Value(indexMY) = dYMiddle
pPolylineFtr.Value(indexMZ) = dZMiddle

'Calculate others
Dim pHeight As Double
Dim pBase As Double
Dim pSlope As Double
pHeight = Abs(dZFrom - dZTo)
pBase = Sqr((dXFrom - dXTo) * (dXFrom - dXTo) + (dYFrom - dYTo) * (dYFrom - dYTo))
pSlope = Atn(pHeight / pBase) * 180 / 3.14159265358979
pPolylineFtr.Value(indexHeight) = pHeight
pPolylineFtr.Value(indexBase) = pBase
pPolylineFtr.Value(indexSlope) = pSlope
pPolylineFtr.Store

Set pPolylineFtr = pPolylineFtrCsr.NextFeature

Wend

End Sub
```

REFERENCES

Ambraseys, N. N. and J. M. Menu (1988). "Earthquake-Induced Ground Displacements." *Earthquake Engineering & Structural Dynamics* **16**(7): 985-1006.

Assatourians, K. and G. M. Atkinson (2010). "Coseismic stress parameter of three California Earthquakes derived from the stochastic finite fault technique." *Journal of Seismology* **14**(3): 431-443.

Atkinson, G. M. and W. Silva (2000). "Stochastic modeling of California ground motions." *Bulletin of the Seismological Society of America* **90**(2): 255-274.

Band, L. E. (1986). "Topographic Partition of Watersheds with Digital Elevation Models." *Water Resources Research* **22**(1): 15-24.

Baum, R. L., D. L. Galloway, et al. (2008). Landslide and land subsidence hazards to pipelines. Reston, Va., U.S. Geological Survey**:** x, 192 p.

Beresnev, I. A. and G. M. Atkinson (1998). "FINSIM: a FORTRAN program for simulating stochastic acceleration time histories from finite faults." *Seism. Res. Lett. 69, 27–32.*

Beresnev, I. A. and G. M. Atkinson (2002). "Source parameters of earthquakes in eastern and western North America based on finite-fault modeling." *Bulletin of the Seismological Society of America* **92**(2): 695-710.

Blake, T. F., R. A. Hollingsworth, et al. (2002). "Recommended Procedures for Implementation of DMG Special Publication 117 Guidelines for Analyzing and Mitigating Landslide Hazards in California." *Southern California Earthquake Center*.

Bommer, J. J., C. E. Rodriguez, et al. (1999). "Earthquake-induced landslides: 1980-1997." *Soil Dynamics and Earthquake Engineering* **18**(5): 325-346.

Boore, D. M. (2000). "SMSIM - Fortran Programs for Simulating Ground Motions from Earthquakes: Version 2.0 A revision of OFR 96-80-A, ." *U.S. Geol. Surv. Open-File Report OF 00-509 (available at http://geopubs.wr.usgs.gov/open-file/of00-509/).*

Boore, D. M. (2003). "Simulation of ground motion using the stochastic method." *Pure and Applied Geophysics* **160**(3-4): 635-676.

Boore, D. M. and G. M. Atkinson (2007). Boore-Atkinson NGA Ground Motion Relations for the Geometric Mean Horizontal Component of Peak and Spectral Ground Motion Parameters, Pacific Earthquake Engineering Research Center.

Boore, D. M. and W. B. Joyner (1997). "Equations for estimating horizontal response spectra and peak acceleration from western north American earthquakes: A summary of recent work " *Seismological Research Letters* **68**(1): 128-153.

Boore, D. M. and W. B. Joyner (1997). "Site amplifications for generic rock sites." *Bulletin of the Seismological Society of America* **87**(2): 327-341.

Boser, B. E., I. M. Guyon, et al. (1992). A training algorithm for optimal margin classifiers. Proceedings of the 5th Annual ACM Workshop on Computational Learning Theory, ACM Press: Pittsburgh, PA.

Box, G. E. P. and N. R. Draper (1986). Empirical model-building and response surface. New York, Wiley.

Bradley, B. A. (2010). "Site-Specific and Spatially Distributed Ground-Motion Prediction of Acceleration Spectrum Intensity." *Bulletin of the Seismological Society of America* **100**(2): 792-801.

Bradley, B. A., M. Cubrinovski, et al. (2009). "Ground-Motion Prediction Equation for SI Based on Spectral Acceleration Equations." *Bulletin of the Seismological Society of America* **99**(1): 277-285.

Braga-Neto, U. M. and E. R. Dougherty (2004). "Is cross-validation valid for small-sample microarray classification?" *Bioinformatics* **20**(3): 374-380.

Bray, J. D. and T. Travasarou (2007). "Simplified procedure for estimating earthquake-induced deviatoric slope displacements." *Journal of Geotechnical and Geoenvironmental Engineering* **133**(4): 381-392.

Brune, J. N. (1970). "Tectonic stress and the spectra of seismic shear waves from earthquakes." *Journal of Geophsyical Research* **75**: 4997-5009.

Burges, C. J. C. (1998). "A tutorial on Support Vector Machines for pattern recognition." *Data Mining and Knowledge Discovery* **2**(2): 121-167.

Campbell, K. W. and Y. Bozorgnia (2007). Campbell-Bozorgnia NGA Ground Motion Relations for the Geometric Mean Horizontal Component of Peak and Spectral Ground Motion Parameters, Pacific Earthquake Engineering Research Center.

Campbell, K. W. and Y. Bozorgnia (2010). "A Ground Motion Prediction Equation for the Horizontal Component of Cumulative Absolute Velocity (CAV) Based on the PEER-NGA Strong Motion Database." *Earthquake Spectra* **26**(3): 635-650.

Carrara, A. and F. Guzzetti (1995). "Geographical information systems in assessing natural hazards." xii, 353 p.

Carrara, A. and R. J. Pike (2008). "GIS technology and models for assessing landslide hazard and risk." *Geomorphology* **94**(3-4): 257-260.

Chang, C.-C. and C.-J. Lin (2001). LIBSVM : a library for support vector machines**:** Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

Chigira, M. and H. Yagi (2006). "Geological and geomorphological characteristics of landslides triggered by the 2004 Mid Niigta prefecture earthquake in Japan." *Engineering Geology* **82**(4): 202-221.

Cho, S. E. (2009). "Probabilistic stability analyses of slopes using the ANN-based response surface." *Computers and Geotechnics* **36**(5): 787-797.

Cole, W. F., P. O. Shires, et al. (1998). "Analysis of Earthquake-Reactivated Landslides in the Epicentral Region, Cental Santa Cruz Mountains, California." *The Loma Prieta, California, Earthquake of October 17, 1989—Landslides, Prof. Paper 1551, USGS, Reston, Va., C71–C128.*

Cortes, C. and V. Vapnik (1995). "Support-Vector Networks." *Machine Learning* **20**(3): 273-297.

Demuth, H. B., M. H. Beale, et al. (2009). Neural Network Toolbox: User's Guide. Natick, Mass., MathWorks, Inc.

Diamantidis, N. A., D. Karlis, et al. (2000). "Unsupervised stratification of cross-validation for accuracy estimation." *Artificial Intelligence* **116**(1-2): 1-16.

Dibike, Y. B., S. Velickov, et al. (2001). "Model induction with support vector machines: Introduction and applications." *Journal of Computing in Civil Engineering* **15**(3): 208-216.

Douglas, J. and H. Aochi (2008). "A Survey of Techniques for Predicting Earthquake Ground Motions for Engineering Purposes." *Surveys in Geophysics* **29**(3): 187-220.

Duda, R. O., P. E. Hart, et al. (2001). Pattern classification. New York, Wiley.

Feng, X. T., H. B. Zhao, et al. (2004). "Modeling non-linear displacement time series of geo-materials using evolutionary support vector machines." *International Journal of Rock Mechanics and Mining Sciences* **41**(7): 1087-1107.

Ferentinou, M. D. and M. G. Sakellariou (2007). "Computational intelligence tools for the prediction of slope performance." *Computers and Geotechnics* **34**(5): 362-384.

Goh, A. T. C. and S. H. Goh (2007). "Support vector machines: Their use in geotechnical engineering as illustrated using seismic liquefaction data." *Computers and Geotechnics* **34**(5): 410-421.

Guzzetti, F., A. Carrara, et al. (1999). "Landslide hazard evaluation: a review of current techniques and their application in a multi-scale study, Central Italy." *Geomorphology* **31**(1-4): 181-216.

Hansen, A. (1984). Lanslide Hazard Analysis. Slope instability. D. Brunsden and D. B. Prior. New York, Wiley**:** 523-602.

Harp, E. L. and R. W. Jibson (1996). "Landslides triggered by the 1994 Northridge, California, earthquake." *Bulletin of the Seismological Society of America* **86**(1): S319-S332.

Harp, E. L., R. C. Wilson, et al. (1981). "Landslides from the February 4, 1976, Guatemala Earthquake." *U.S. Geological Survey Professional Paper 1204-A.*

Haykin, S. S. (1999). Neural networks : a comprehensive foundation. Upper Saddle River, N.J., Prentice Hall.

Holzer, T. L., M. J. Bennett, et al. (2005). "Shear-wave velocity of surficial geologic sediments in northern California: Statistical distributions and depth dependence." *Earthquake Spectra* **21**(1): 161-177.

Hopgood, A. A. (2001). Intelligent systems for engineers and scientists. Boca Raton, Fla., CRC Press.

Itasca (2008). "FLAC-Fast Lagrangian Analysis of Continua-User's Guide."

ITASCA, C. G. I. (2008). FLAC fast Lagrangian analysis of continua, Minneapolis, Minnesota (USA).

J. Chiou, B. S. and R. R. Youngs (2008). NGA Model for Average Horizontal Component of Peak Ground Motion and Response Spectra, Pacific Earthquake Engineering Research Center.

Jain, A. K., R. P. W. Duin, et al. (2000). "Statistical pattern recognition: A review." *Ieee Transactions on Pattern Analysis and Machine Intelligence* **22**(1): 4-37.

Jenson, S. K. and J. O. Domingue (1988). "Extracting Topographic Structure from Digital Elevation Data for Geographic Information-System Analysis." *Photogrammetric Engineering and Remote Sensing* **54**(11): 1593-1600.

Jibson, R. W. (1993). "Predicting Earthquake-induced Landslide Displacements Using Newmark's Sliding Block Analysis." *Transportation Research Record 1411, National Research Council, Washington, D.C., 9–17.*

Jibson, R. W. (2007). "Regression models for estimating coseismic landslide displacement." *Engineering Geology* **91**(2-4): 209-218.

Jibson, R. W., A. J. Crone, et al. (2004). "Landslides triggered by the 13 January and 13 February 2001 earthquakes in El Salvador." *Geological Society of America Special Papers* **375**: 69-88.

Jibson, R. W., E. L. Harp, et al. (2000). "A method for producing digital probabilistic seismic landslide hazard maps." *Engineering Geology* **58**(3-4): 271-289.

Jibson, R. W. and D. K. Keefer (1993). "Analysis of the Seismic Origin of Landslides - Examples from the New-Madrid Seismic Zone." *Geological Society of America Bulletin* **105**(4): 521-536.

Joseph, V. R. and Y. Hung (2008). "Orthogonal-maximin Latin hypercube designs." *Statistica Sinica* **18**(1): 171-186.

Kecman, V. (2001). Learning and soft computing : support vector machines, neural networks, and fuzzy logic models. Cambridge, Mass., MIT Press.

Keefer, D. K. (1984). "Landslides Caused by Earthquakes." *Geological Society of America Bulletin* **95**(4): 406-421.

Keefer, D. K. (1998). "The Loma Prieta, California, Earthquake of October 17, 1989 – Landslides." *U.S. Geological Survey Professional Paper 1551-C.*

Keefer, D. K. (2000). "Statistical analysis of an earthquake-induced landslide distribution - the 1989 Loma Prieta, California event." *Engineering Geology* **58**(3-4): 231-249.

Keefer, D. K. (2002). "Investigating landslides caused by earthquakes - A historical review." *Surveys in Geophysics* **23**(6): 473-510.

Keefer, D. K., ed., (1991). "Geologic Hazards in the Summit Ridge Area of the Santa Cruz Mountains, Santa Cruz County, California, Evaluated in Response to the October 17, 1989, Loma Prieta Earthquake: Report if the Technical Advisory Group." *U.S. Geological Survey Open-File Report 91-618, 427 p.*

Keefer, D. K., G. B. Griggs, et al. (1998). "Large landslides near the San Andreas fault in the Summit Ridge area Santa Cruz Mountains, California." *The Loma Prieta, California, Earthquake of October 17, 1989—Landslides, Prof. Paper 1551, USGS, Reston, Va., C71–C128.*

Keefer, D. K. and M. C. Larsen (2007). "Assessing landslide hazards." *Science* **316**(5828): 1136-1138.

Keefer, D. K. and W. M. Manson (1998). "Regional Distribution and Characteristics of Landslides Generated by the Earthquake." *The Loma Prieta, California, Earthquake of October 17, 1989—Landslides, Prof. Paper 1551, USGS, Reston, Va., C7.*

Khazai, B. (2004). GIS Approach to Seismic Slope Stability. Civil Engineering. Berkeley, University of California Berkeley. Doctor of Philosophy.

Khazai, B. and N. Sitar (2004). "Evaluation of factors controlling earthquake-induced landslides caused by Chi-Chi earthquake and comparison with the Northridge and Loma Prieta events." *Engineering Geology* **71**(1-2): 79-95.

Kim, D. H., D. J. Kim, et al. (1999). "The application of neural networks and statistical methods to process design in metal forming processes." *International Journal of Advanced Manufacturing Technology* **15**(12): 886-894.

Kim, J. H. (2009). "Estimating classification error rate: Repeated cross-validation, repeated hold-out and bootstrap." *Computational Statistics & Data Analysis* **53**(11): 3735-3745.

Kobayashi, Y. (1981). "Causes of Fatalities in Recent Earthquakes in Japan." *Journal of Disaster Sciences, 3, 15-33.*

Kramer, S. L. (1996). Geotechnical earthquake engineering. Upper Saddle River, N.J., Prentice Hall.

Lagaros, N. D. and Y. Tsompanakis (2007). Intelligent computational paradigms in earthquake engineering. Hershey, Idea Group Pub.

Lam, C. Q. (2008). Sequential Adaptive Designs in Computer Experiments for Response Surface Model Fit. Department of Statistics, The Ohio State University.

Lin, J. S. and R. V. Whitman (1983). "Decoupling Approximation to the Evaluation of Earthquake-Induced Plastic Slip in Earth Dams." *Earthquake Engineering & Structural Dynamics* **11**(5): 667-678.

Luo, H. (2006). GIS-Based Approaches to Slope Stability Analysis and Earthquake-Induced Landslide Hazard Zonation. Fairbanks, Alaska, University of Alaska Fairbanks. Doctor of Philosophy.

Luzi, L. and F. Pergalani (1996). "Applications of statistical and GIS techniques to slope instability zonation (1:50.000 Fabriano geological map sheet)." *Soil Dynamics and Earthquake Engineering* **15**(2): 83-94.

Mahdavifar, M. R., S. Solaymani, et al. (2006). "Landslides triggered by the Avaj, Iran earthquake of June 22, 2002." *Engineering Geology* **86**(2-3): 166-182.

Maidment, D. R. (2002). Arc hydro : GIS for water resources. Redlands, Calif., ESRI Press.

Makdisi, F. I. and H. B. Seed (1978). "Simplified Procedure for Estimating Dam and Embankment Earthquake-Induced Deformations." *Journal of the Geotechnical Engineering Division-Asce* **104**(7): 849-867.

Mankelow, J. M. (1997). Probabilistic Techniques for Estimating Earthquake Triggered Landslide Hazards. Department of Geology, University of Portsmouth United Kingdom. Doctor of Philosophy.

McCrink, T. P. and C. R. Real (1996). Evaluation of the Newmark method for mapping earthquake-induced landslide hazards in the Laurel 7.5' Quadrangle, Santa Cruz County, California. Final Technical Report to the US Geological Survey, Award No. 1434-93-G 2334, 32 pp. .

Mckay, M. D., R. J. Beckman, et al. (1979). "A comparison of three methods for selecting values of input variables in the analysis of output from a computer code." *Technometrics* **21**: 239-245.

McLaughlin, R. J., J. C. Clark, et al. (2002). Geologic Maps and Structure Sections of the southwestern Santa Clara Valley and southern Santa Cruz Mountains, Santa Clara and Santa Cruz Counties, California. U.S. Geological Survey Miscellaneous Field Studies 2373.

Michie, D., D. J. Spiegelhalter, et al. (1994). Machine learning, neural and statistical classification. New York ; London, Ellis Horwood.

Miles, S. B. and C. L. Ho (1999). "Rigorous landslide hazard zonation using Newmark's method and stochastic ground motion simulation." *Soil Dynamics and Earthquake Engineering* **18**(4): 305-323.

Miles, S. B. and D. K. Keefer (2009). "Evaluation of CAMEL - comprehensive areal model of earthquake-induced landslides." *Engineering Geology* **104**(1-2): 1-15.

Miller, D. J. (1995). "Coupling GIS with physical models to assess deep-seated landslide hazards." *Environmental and Engineering Geoscience* **1**(3): 263-276.

Molinaro, A. M., R. Simon, et al. (2005). "Prediction error estimation: a comparison of resampling methods." *Bioinformatics* **21**(15): 3301-3307.

Morris, M. D. and T. J. Mitchell (1995). "Exploratory Designs for Computational Experiments." *Journal of Statistical Planning and Inference* **43**(3): 381-402.

Motazedian, D. and G. M. Atkinson (2005). "Stochastic finite-fault modeling based on a dynamic corner frequency." *Bulletin of the Seismological Society of America* **95**(3): 995-1010.

Myers, R. H. and D. C. Montgomery (1995). Response surface methodology : process and product optimization using designed experiments. New York, Wiley.

Newmark, N. M. (1965). "Effects of Earthquakes on Dams and Embankments." *Geotechnique* **15**(2): 139.

Owen, A. B. (1994). "Controlling Correlations in Latin Hypercube Samples." *Journal of the American Statistical Association* **89**(428): 1517-1522.

Paolucci, R., P. Colli, et al. (2000). "Assessment of Seismic Site Effects in 2-D Alluvial Valleys Using Neural Networks." *Earthquake Spectra* **16**(3): 661-680.

Parise, M. and R. W. Jibson (2000). "A seismic landslide susceptibility rating of geologic units based on analysis of characteristics of landslides triggered by the 17 January, 1994 Northridge, California earthquake." *Engineering Geology* **58**(3-4): 251-270.

Park, J. S. (1994). "Optimal Latin-Hypercube Designs for Computer Experiments." *Journal of Statistical Planning and Inference* **39**(1): 95-111.

Power, M., B. Chiou, et al. (2008). "An overview of the NGA project." *Earthquake Spectra* **24**(1): 3-21.

Rathje, E. M., N. A. Abrahamson, et al. (1998). "Simplified frequency content estimates of earthquake ground motions." *Journal of Geotechnical and Geoenvironmental Engineering* **124**(2): 150-159.

Rathje, E. M. and J. D. Bray (1999). "An examination of simplified earthquake-induced displacement procedures for earth structures." *Canadian Geotechnical Journal* **36**(1): 72-87.

Rathje, E. M., F. Faraj, et al. (2004). "Empirical relationships for frequency content parameters of earthquake ground motions." *Earthquake Spectra* **20**(1): 119-144.

Refice, A. and D. Capolongo (2002). "Probabilistic modeling of uncertainties in earthquake-induced landslide hazard assessment." *Computers & Geosciences* **28**(6): 735-749.

Roberts, S. and A. D. Baron (1998). Digital Compilation of "Preliminary Map of Landslide Deposits in Santa Cruz County, California, By Cooper-Clark and Associates, 1975": A Digital Map Database, U.S. Geological Survey.

Santner, T. J., B. J. Williams, et al. (2003). The Design and analysis of computer experiments. New York, Springer.

Saygili, G. and E. M. Rathje (2008). "Empirical predictive models for earthquake-induced sliding displacements of slopes." *Journal of Geotechnical and Geoenvironmental Engineering* **134**(6): 790-803.

Scholkopf, B., A. J. Smola, et al. (2000). "New support vector algorithms." *Neural Computation* **12**(5): 1207-1245.

Seed, H. B. and G. R. Martin (1966). "The seismic coefficient in earth dam design." *Journal of the Soil Mechanics and Foundations Division, ASCE, 92(SM3): 25-58.*

Shunmugam, M. S. and N. S. Prasad (2008). "Prediction of stress in fillet portion of spur gears using artificial neural networks." *Ai Edam-Artificial Intelligence for Engineering Design Analysis and Manufacturing* **22**(1): 41-51.

Simpson, T. W. (1998). A Concept Exploration Method for Product Family Design. Mechanical Engineering, Georgia Institute of Technology. Doctor of Philosophy.

Simpson, T. W., J. D. Peplinski, et al. (2001). "Metamodels for computer-based engineering design: survey and recommendations." *Engineering with Computers* **17**(2): 129-150.

Smola, A. J. and B. Scholkopf (2004). "A tutorial on support vector regression." *Statistics and Computing* **14**(3): 199-222.

Strenk, P. M. (2010). Evaluation of Analytical Procedures for Estimating Seismically Induced Permanent Deformations in Slopes, Drexel University. Doctor of Philosophy.

Tang, B. X. (1998). "Selecting Latin hypercubes using correlation criteria." *Statistica Sinica* **8**(3): 965-977.

Towashiraporn, P. (2004). Building Seismic Fragilities Using Response Surface Metamodels. Civil and Environmental Engineering, Georgia Institute of Technology.

Turner, A. K. and R. L. Schuster (1996). Landslides : investigation and mitigation. Washington, DC, National Academy Press.

Vapnik, V. N. (1995). The nature of statistical learning theory. New York, Springer.

Vapnik, V. N. (1998). Statistical learning theory. New York, Wiley.

Varnes, D. J. (1978). "Slope Movement Types and Processes." *In Special Report 176: Landslides: Analysis and Control (R.L. Schuster and R.J. Krizek, eds.), TRB, National Research Council, Washington, D.C., pp.11-33*.

Wald, D. J., D. V. Helmberger, et al. (1991). "Rupture Model of the 1989 Loma-Prieta Earthquake from the Inversion of Strong-Motion and Broad-Band Teleseismic Data." *Bulletin of the Seismological Society of America* **81**(5): 1540-1572.

Wang, F. W., Q. G. Cheng, et al. (2009). "Preliminary investigation of some large landslides triggered by the 2008 Wenchuan earthquake, Sichuan Province, China." *Landslides* **6**(1): 47-54.

Wang, F. W., Y. P. Yin, et al. (2009). "Landslide hazards triggered by the 2008 Wenchuan earthquake, Sichuan, China." *Landslides* **6**(2): 139-152.

WG (2003). Earthquake Probabilities in the San Francisco Bay Region: 2002–2031. USGS Open-File Report 03-214.

Wieczorek, G. F., R. C. Wilson, et al. (1985). Map showing stability during earthquakes in San Mateo County California. US Geol. Surv. Misc. Invest. Map I-1257-E, Scale 1:62,500.

Wills, C. J. and K. B. Clahan (2006). "Developing a map of geologically defined site-condition categories for California." *Bulletin of the Seismological Society of America* **96**(4): 1483-1501.

Wills, C. J., M. Petersen, et al. (2000). "A site-conditions map for California based on geology and shear-wave velocity." *Bulletin of the Seismological Society of America* **90**(6): S187-S208.

Wilson, R. I., T. P. McCrink, et al. (2008). "ShakeOut Scenario Appendix F: Seismically Induced Landslide Hazard Analysis and Deformation at Lifeline Crossings." *USGS Open File Report 2008-1150, Appendix F CGS Preliminary Report 25F*.

Xie, M. W., T. Esaki, et al. (2003). "Geographic information systems-based three-dimensional critical slope stability analysis and landslide hazard assessment." *Journal of Geotechnical and Geoenvironmental Engineering* **129**(12): 1109-1118.

Zienkiewicz, O., C. Humpheson, et al. (1975). "Associated and nonassociated viscoplasticity and plasticity in soil mechanics." *Géotechnique* **25**(4): 671-689.